



DESMOND BALA BISANDU

NOVEL DEEP LEARNING APPROACHES FOR FLIGHT DELAYS
PREDICTION

SCHOOL OF AEROSPACE, TRANSPORT AND
MANUFACTURING
(COMPUTATIONAL ENGINEERING SCIENCES)

DOCTOR OF PHILOSOPHY (PhD)
Academic Year: 2019 - 2023

Supervisor: Dr Irene Moulitsas
Associate Supervisor: Dr Salvatore Filippone
April 2023



SCHOOL OF AEROSPACE, TRANSPORT AND
MANUFACTURING
(COMPUTATIONAL ENGINEERING SCIENCES)

DOCTOR OF PHILOSOPHY (PhD)

Academic Year 2019 - 2023

DESMOND BALA BISANDU

Novel Deep Learning approaches for flight delays prediction

Supervisor: Dr Irene Moulitsas
Associate Supervisor: Dr Salvatore Filippone
April 2023

This thesis is submitted in partial fulfilment of the requirements for
the degree of Doctor of Philosophy

© Cranfield University 2023. All rights reserved. No part of this
publication may be reproduced without the written permission of the
copyright owner.

"In machine learning, the real magic happens when algorithms learn from data, adapt, and improve over time."

Yann LeCun

"Intelligence is the ability to adapt to change."

Stephen Hawking

"If a machine is expected to be infallible, it cannot also be intelligent."

Alan Turing

ABSTRACT

This thesis focuses on the development of predictive models for flight delay using the United States (US) Bureau of Transportation Statistics (BTS) dataset. The research aims to assess the potential of the BTS data for creating data-driven flight delay predictive models. Exploratory Data Analysis (EDA) reveals complex relationships, irregularities, outliers, missing values and invalid values within the dataset, posing challenges for traditional Machine Learning algorithms and resulting in low-performance models.

To address these limitations, six advanced prediction algorithms are developed. The approaches include binary classification using a filter-based procedure and deep-stacked auto-encoder, a hybrid ensemble learner, a fully connected Bi-Long-Short Term Memory (LSTM), a social ski driver conditional autoregressive value at risk optimisation-aware deep neural network, and a novel gradient mayfly-based optimisation DeepONet approach.

The US BTS dataset, along with expert-defined records, is used to validate the proposed algorithms. The research findings demonstrate that an optimisation-aware data-driven predictive approach yields reliable and robust results for flight delay prediction. Among the developed approaches, the gradient mayfly optimisation DeepONet model, particularly the proposed data-driven optimisation-aware DeepONet, outperforms other strategies in terms of prediction error on benchmark evaluation metrics.

The validation results highlight the effectiveness of the data-driven optimisation-aware DeepONet model, achieving a minimum prediction error of 0.0043. This suggests that more than 90% of flights can be accurately predicted based on the findings of this research.

Keywords:

Airline flight delays classification, air transport system, DeepONet, optimisation algorithms, Machine Learning, predictive modelling.

ACKNOWLEDGEMENTS

I dedicated this work to my entire family for their support and, most importantly, to God almighty for the strength throughout my studies.

I want to thank my PhD supervisor, Dr Irene Moulitsas, who always inspires me to focus on the bigger goal. I am fortunate to have her as a supervisor who is always ready to support feedback and queries and correct me when I am wrong. Her wisdom in dealing with students is incomparable and second to none. I sincerely appreciate her consistently checking my progress and encouraging me, especially being the only supervisor, I could ever want.

I want to acknowledge the Dr Salvatore Filippone, my associate advisor, but unfortunately, we could not spend a long time before he left for another school in Italy. I appreciate all the support you gave me during our short time together; it was worth a while.

My gratitude extends to the Head of the Centre for Computational Engineering, Professor Karl W. Jenkins. Thank you for always approving my supervisor's request to make me grow professionally by attending conferences and workshops nationally and internationally.

I would also like to thank all the staff in the Digital Aviation Research Technology Centre (DARTEC), especially Dr Thomas Budd and Dr Darren Ellis, for accepting to do my annual reviews during my studies and my colleagues in the Machine Learning and data analytics laboratory, (DARTEC). Thank you for all the collaboration and time spent together as good friends.

Finally, I cannot thank enough my wife (Priscilla Bisandu) and my twins' children (Alisiya Dalziel and Kasili Darrell) for their constant support throughout my PhD study.

To Cranfield University and the Petroleum Technology Development Funds (PTDF), Nigeria, this thesis would not have been possible without your funding and enabling environment for my research.

TABLE OF CONTENTS

ABSTRACT	i
ACKNOWLEDGEMENTS.....	ii
TABLE OF CONTENTS	iii
LIST OF FIGURES.....	vii
LIST OF TABLES	x
LIST OF EQUATIONS.....	xii
LIST OF ABBREVIATIONS	xvii
Chapter 1: General Introduction	1
1.1 Motivation for the Study	2
1.2 Aim and Objectives	4
1.2.1 Aim.....	4
1.2.2 Objectives	4
1.3 List of Publications	5
1.4 General Research Methodology and Thesis Structure.....	6
1.5 Thesis Presentation	7
1.6 References.....	11
Chapter 2: A Systematic Literature Review on Flight Delays Prediction	
Methods	14
2.1 Introduction	14
2.2 A Data Scientist Perspective of Air Passenger Experience	16
2.3 A Taxonomic Analysis of Passenger Experience Problem	18
2.3.1 Problem.....	22
2.3.2 Scope	26
2.3.3 Data.....	27
2.3.4 Methods	31
2.4 Discussion of Results.....	36
2.5 Conclusion, Summary of Findings and Recommendations.....	40
2.6 Research Gaps.....	42
2.7 References	42
Chapter 3: Exploratory Data Analysis: A Flight On-Time Dataset View.....	66
3.1 Introduction	66
3.1.1 Deriving Intelligence from Data	67
3.1.2 The Problem.....	68
3.2 Related Work	69
3.3 Datasets Description.....	70
3.4 Approach for Analysis	73
3.5 Visualisation and Discussion	74
3.6 Limitations and Future Directions	83
3.7 Conclusion	83
3.8 References	85

Chapter 4: A Deep Feedforward Neural Network and Shallow Architectures	
Effectiveness Comparison: Flight Delays Classification Perspective	91
4.1 Introduction	92
4.2 Related Work	94
4.3 Methods	96
4.3.1 Support Vector Machine	96
4.3.2 Neural Network	97
4.3.3 Deep Feedforward Neural Network	98
4.4 Discussion of Results	101
4.4.1 Experimental Setup	102
4.4.2 Results	105
4.5 Conclusion and Future Direction	111
4.6 References	113
Chapter 5: An Enhanced Deep Autoencoder for Flight Delay Prediction	118
5.1 Introduction	118
5.2 Related Work	120
5.3 Materials and Methods	124
5.3.1 Proposed Architecture	127
5.3.2 Experimentation	128
5.3.3 Dataset Collection and Description	129
5.3.4 Evaluation Metrics	131
5.4 Results and Discussion	132
5.4.1 Vanilla Autoencoder	132
5.4.2 Logistic Regression Autoencoder	135
5.4.3 Multilayer Perceptron Autoencoder	137
5.4.4 Comparative Discussion	138
5.5 Conclusion and Future Direction	139
5.6 References	141
Chapter 6: A Hybrid Ensemble Machine Learning Approach for Arrival Flight	
Delay Classification Prediction Using Voting Aggregation Technique	147
6.1 Introduction	147
6.2 Related Works	149
6.3 Materials and Methods	151
6.3.1 Feature Characterisation	152
6.3.2 Sub-sampling Method	154
6.3.3 Support Vector Classifiers	155
6.4 Results and Discussion	161
6.5 Conclusion	166
6.6 References	168
Chapter 7: A Deep Bi-LSTM Machine Learning Method for Flight Delay	
Prediction Classification	172
7.1 Introduction	172

7.2 Related Work	175
7.3 Materials and Methods.....	177
7.3.1 Data Description and Pre-processing.....	178
7.3.2 Proposed methodology	181
7.3.3 Model Evaluation Metrics	186
7.4 Results and Discussion.....	187
7.4.1 Experimentation Setup.....	188
7.4.2 Results Comparative Discussions.....	189
7.5 Conclusion and Future Direction.....	191
7.6 References	193
Chapter 8: Social Ski Driver Conditional Autoregressive-Based Deep Learning Classifier for Flight Delay Prediction.....	197
8.1 Introduction	197
8.2 Related Literature	200
8.3 Problem Statement	204
8.4 Flight Delay Prediction using the Proposed Social Ski Driver Conditional Autoregressive-based (SSDCA) Deep Learning classifier	205
8.4.1 Dataset Description.....	206
8.4.2 Pre-processing	208
8.4.3 Data Transformation using Yeo-Johnson Transformation.....	209
8.4.4 Feature Fusion using Proposed SSDCA-based Deep RNN.....	210
8.4.5 Deep LSTM Architecture	218
8.5 Results and Discussion.....	221
8.5.1 Dataset Source	221
8.5.2 Evaluation Metrics	222
8.5.3 Method Comparisons	222
8.5.4 Comparative Discussion.....	234
8.5.5 Statistical Analysis.....	235
8.5.6 Computational Time Analysis.....	235
8.6 Conclusions and Future Work.....	236
8.7 References	238
Chapter 9: Prediction of Flight Delay using Deep Operator Network with Gradient-Mayfly Optimisation Algorithm	244
9.1 Introduction	244
9.2 Literature Review	247
9.2.1 Problem Definition	249
9.3 Proposed GMOA-based DeepONet for Prediction of Flight Delay	250
9.3.1 Pre-processing for Noise Removal.....	251
9.3.2 Data Transformation with Box cox Transformation	251
9.4 Results and Discussion.....	264
9.4.1 Dataset Description	264
9.4.2 Performance Evaluation Measure	265

9.4.3 Comparison of Techniques	266
9.4.4 Comparative Analysis.....	266
9.4.5 Comparative Discussion of the Techniques	275
9.5 Conclusion and Lesson Learned	277
9.6 References	279
Chapter 10: General Discussion and Findings	285
10.1 A Deep Feedforward Neural Network Algorithm for Delay Prediction and Analysis	286
10.2 An Enhanced Stacked Autoencoder Algorithm for Flight Delay Prediction.....	286
10.3 A Hybrid Voting Ensemble Approach for Flight Delay Classification...	287
10.4 A Deep Bi-LSTM Approach for Flight Delay Modelling and Simulation	287
10.5 A Social Ski Driver Conditional Auto-regressive Deep Learning for Flight Delay Prediction	288
10.6 A Mayflies Optimisation Driven DeepONet Approach for Flight Delay Prediction.....	289
10.7 A Comparative Analysis of the proposed Methods and Generalisation	290
10.8 References	292
Chapter 11: Conclusion and Future Work	295
11.1 How the Research Aim and Objectives Were Answered	296
11.2 Contribution to Knowledge.....	299
11.3 Intellectual Contribution and its Impact	301
11.4 Limitation of the Research	302
11.5 Future Work	302

LIST OF FIGURES

Figure 1-2 An Overview of the General PhD Research Methodology	6
Figure 2-1 A traditional Commercial Flight Operation.....	18
Figure 2-2 Taxonomy of Passenger Experience Problem	20
Figure 2-3 Procedure for Snowballing Search.....	20
Figure 2-4 Passenger Experience Data Model.....	30
Figure 2-5 Categories of Methods for the Passenger Experience Model	32
Figure 2-6 Relationship Between the Year and Passenger Experience Research Articles	37
Figure 2-7a Publication on Main Methods According to Years	38
Figure 2-7b Methods Application to Problems.....	39
Figure 3-1 An Exploratory Data Framework	68
Figure 3-2 Statistics of Delays Among Unique Carriers	75
Figure 3-3 Correlation Matrix of Attributes.....	76
Figure 3-4 Mean Delays of Carriers	77
Figure 3-5 Arrival Delay Distribution of Carriers	77
Figure 3.6 Flights Counts of Origin Airports.....	78
Figure 3-7 Box Plot of Delays in a Week.....	78
Figure 3-8 Total Number of Flights by Airline	79
Figure 3-9 Total Number of Flights Delayed by Airlines	79
Figure 3-10 Airline Flight Delay Percentage.....	80
Figure 3-11 Most Popular Destinations - Arrival Delay.....	81
Figure 3-12 Most Popular Cities with Departure Delays.....	81
Figure 3-13 Number of Destinations by Airline	82
Figure 3-14 Top Airports & Average Delay.....	82
Figure 4-1 Architecture of Deep Feedforward Neural Network.....	100
Figure 4-2 An Overview of the Experiment Steps.....	102
Figure 4-3 Confusion Matrix of Testing Classification using DNN with an Epoch=500	108
Figure 4-4 Confusion Matrix of Testing Classification using DNN with an Epoch=200	109
Figure 4-5 Model Accuracy Plot for DNN with Epoch=500	110
Figure 4-6 Model Loss Plot for DNN with Epoch=500	110
Figure 4-7 Final DNN Model Architecture.....	111

Figure 5-1 Structure of Autoencoders	125
Figure 5-2 Sigmoid Activation Function	126
Figure 5-3 The Proposed Architecture	127
Figure 5-4 multilayer perceptron Autoencoder	128
Figure 5-5 Logistic Regression Autoencoder.....	128
Figure 5-6 Vanilla Autoencoder	129
Figure 5-7 Precision-Recall Curve for 10 Layers and 300 Epochs	133
Figure 5-8 Confusion Matrix for 10 Layers and 300 Epochs	134
Figure 5-9 Precision and Recall for Different Threshold Values	134
Figure 5-10 Confusion Matrix for 11 Layers and 300 Epochs	136
Figure 5-11 Model Loss for 17 Layers and 300 Epochs	136
Figure 5-12 Precision-Recall Curve for 7 Layers and 30 Epochs	137
Figure 5-13 Confusion Matrix for 7 Layers and 30 Epochs	138
Figure 6-1 The Proposed Ensemble Method.....	152
Figure 6-2 ROC Curve of Models	162
Figure 6-3 Prediction Results Plot of Different Methods	164
Figure 7-1 Proposed Prediction Methodology	182
Figure 7-2 An Architecture of a BiLSTM.....	184
Figure 7-3 An Architecture of a Unidirectional LSTM	185
Figure 7-4 The Structure of a Confusion Matrix for a Binary Classification Task	1875
Figure 7-5 Flight Correlation Matrix	186
Figure 7-6 Confusion Matrix for LSTM approach.....	187
Figure 7-7 Confusion Matrix for BiLSTM approach	188
Figure 8-1 Block diagram of the Flight Delay Prediction using the Proposed SSDCA-based Deep Learning Classifier	206
Figure 8-2 Schematic Diagram for the Determining-processing Output	209
Figure 8-3 Structure of Deep RNN	212
Figure 8-4 Architecture of Deep LSTM Classifier	219
Figure 8-5 Analysis using Dataset-1 with feature size 6, a) RMSE, b) MSE, c) MAE, d) Accuracy.....	226
Figure 8-6 Analysis using Dataset-2 with feature size 6, a) RMSE, b) MSE, c) MAE, d) Accuracy.....	230
Figure 8-7 Delay Prediction using Dataset-1.....	231
Figure 8-8 Delay Prediction using Dataset-2	232

Figure 8-9 Convergence Analysis using Dataset-1	233
Figure 10 Convergence Analysis using Dataset-2.....	233
Figure 9-1 Structure of Flight Delay Prediction Model using Proposed Gmoa- Based DeepONet	250
Figure 9-2 Architectural Model of DRN.....	253
Figure 9-3 Schematic View of DeepONet.....	259
Figure 9-4 Assessment of Techniques using a) RMSE, b) MSE, c) MAE and d) MAPE	270
Figure 9-5 Assessment of Techniques using a) RMSE, b) MSE, c) MAE and d) MAPE	274
Figure 9-6 Analysis of Techniques using a) Jan 2021 data b) Nov 2021 data	275

LIST OF TABLES

Table 1-1 Summary of US departures on-time performance from 2013 to 2022	3
Table 1-2 Summary of US arrivals on-time performance from 2013 to 2022.....	4
Table1-3 List of publications.....	5
Table 2-1 A summary result of extracted papers for analysis.....	21
Table 2-2 Selected input features for this thesis and other expert opinion selected features in literature.....	25
Table 2-3 Air transportation system real data sources per region	28
Table 3-1 Airports dataset information	71
Table 3-2 Dataset information	71
Table 4-1 Parameter settings for SVM, NN and DNN	101
Table 4-2 Software environment information.....	102
Table 4-3 Dataset factors information	103
Table 4-4 Performance comparison of SVM, NN and DNN on flight delays prediction using BTS dataset.....	106
Table 4-5 Performance comparison of DNN on flight delays prediction using BTS dataset with different settings	107
Table 4-6 Performance comparison of DNN on flight delays prediction using BTS dataset with different settings and 500 epochs.....	108
Table 5-1 Dataset features	130
Table 5-2 Libraries	131
Table 5-3 Performance comparison of vanilla autoencoder	133
Table 5-4 Performance comparison for logistic regression	135
Table 5-5 Performance comparison for multilayer perceptron.....	137
Table 5-6 Performance comparisons of the different autoencoders	139
Table 6-1 Dataset features and description.....	159
Table 6-2 The Confusion matrix	160
Table 6-3 Special names for ensembles	163
Table 6-4 Prediction results of different methods	163
Table 6-5 Confusion matrix results of the ensemble Machine Learning	165
Table 6-6 Results of bagging ensemble models with 50-fold cross-validation	166
Table 7-1 Dataset features and their descriptions	178
Table 7-2 Features used in training our model	180
Table 7-3 Airports codes	181
Table 7-4 Performance comparison of LSTM and BiLSTM models.....	189

Table 8-1 Datasets information	207
Table 8-2 Determination of training data	214
Table 8-3 Methods	221
Table 8-4 Results comparison of our proposed method with other existing methods across different evaluation metrics on dataset-1	224
Table 8-5 Results comparison of our proposed method with other existing methods across different evaluation metrics on dataset-2	228
Table 8-6 Delay prediction using dataset-1	231
Table 8-7 Delay prediction using dataset-2	232
Table 8-8 Comparative discussion	235
Table 8-9 Statistical analysis	235
Table 8-10 Computational time	236
Table 9-1 Discovery of training data.....	255
Table 9-2 Special name for methods.....	266
Table 9-3 Results comparison of the proposed method and benchmark methods on various evaluation metric.....	267
Table 9-4 Results comparison of the proposed method and benchmark methods on various evaluation metric.....	272
Table 9-5 Comparative assessment.....	276

LIST OF EQUATIONS

(2-1).....36

(2-2).....36

(3-1).....72

(3-2).....72

(3-3).....72

(4-1).....97

(4-2)..... 98

(4-3)..... 99

(4-4)..... 99

(4-5)..... 99

(4-6)..... 99

(4-7)..... 100

(4-8).....105

(4-9)..... 105

(4-10)..... 105

(4-11)..... 105

(4-12)..... 105

(5-1).....125

(5-2).....125

(5-3).....126

(5-4).....126

(5-5).....126

(5-6)..... 126

(5-7).....131

(5-8).....131

(5-9).....131

(6-1).....153

(6-2).....153

(6-3).....153

(6-4).....153

(6-5).....153

(6-6).....153

(6-7).....154

(6-8).....	154
(6-9).....	154
(6-10).....	155
(6-11).....	155
(6-12).....	155
(6-13).....	157
(6-14).....	160
(6-15).....	160
(6-16).....	160
(6-17).....	160
(6-18).....	160
(6-19).....	160
(7-1).....	181
(7-2).....	182
(7-3).....	182
(7-4).....	182
(7-5).....	182
(7-6).....	183
(7-7).....	183
(7-8).....	183
(7-9).....	186
(7-10).....	186
(7-11).....	186
(7-12).....	186
(7-13).....	187
(8-1).....	205
(8-2).....	205
(8-3).....	209
(8-4).....	210
(8-5).....	210
(8-6).....	210
(8-7).....	210
(8-8).....	210
(8-9).....	210

(8-10).....	214
(8-11).....	215
(8-12).....	215
(8-13).....	212
(8-14).....	215
(8-15).....	215
(8-16).....	215
(8-17).....	2143
(8-18).....	2153
(8-19).....	2153
(8-20).....	213
(8-21).....	2153
(8-22).....	2153
(8-23).....	2153
(8-24).....	2143
(8-25).....	2154
(8-26).....	2156
(8-27).....	217
(8-28).....	2157
(8-29).....	2157
(8-30).....	2157
(8-31).....	2147
(8-32).....	2157

(8-33).....	2158
(9-1).....	250
(9-2).....	251
(9-3).....	252
(9-4).....	252
(9-5).....	253
(9-6).....	253
(9-7).....	254
(9-8).....	254
(9-9).....	254
(9-10).....	254
(9-11).....	254
(9-12).....	255
(9-13).....	255
(9-14).....	256
(9-15).....	256
(9-16).....	256
(9-17).....	256
(9-18).....	256
(9-19).....	256
(9-20).....	256
(9-21).....	256
(9-22).....	257
(9-23).....	257
(9-24).....	257
(9-25).....	257
(9-26).....	257
(9-27).....	259
(9-28).....	260
(9-29).....	258
(9-30).....	2568
(9-31).....	2568

(9-32).....	2568
(9-33).....	2568
(9-34).....	2568
(9-35).....	2569
(9-36).....	2569
(9-37).....	2569
(9-38).....	2579
(9-39).....	2579
(9-40).....	2579
(9-41).....	2579
(9-42).....	2579
(9-43).....	2599
(9-44).....	2609
(9-45).....	26060
(9-46).....	263
(9-47).....	263
(9-48).....	263
(9-49).....	263
(9-50).....	265
(9-51).....	265
(9-52).....	265

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimisation
AERFOR	Airport with an Advanced Forecasting System
AFP	Airport of the Future Project
ANN	Artificial Neural Network
ATMB	Air Traffic Management Bureau
ATS	Air Transportation Systems
BH	Baggage Handling
BTS	Bureau of Transportation Statistics
CAA	Civil Aviation Administration
CDM	Collaborative Decision Making
DL	Deep Learning
DNN	Deep Neural Network
DRNN	Deep Recurrent Neural Network
EDA	Exploratory Data Analysis
ETL	Extract, Transform and Load
EWA	Earthworm Optimisation Algorithm
FAA	Federal Aviation Agency
FAA	Federal Aviation Administration
FDP	Flight Delay Predictions
FN	False Negative
FOTP	Flight On-time Performance
FP	False Positive
GDP	Ground Delay Programs
GMM-EM	Gaussian Mixture Model and Expectation Maximisation
HBO	Honey-Bee Optimisation
IATA	International Airport Transport Authority
ICAO	International Civil Aviation Organisation

IoT	Internet of Things
LightGBM	Light Gradient Boosting Machine
LM	Levenberg-Marquart
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MATNFO	Multi-Objective Air Traffic Network Flow Optimisation
ML	Machine Learning
MSE	Mean Square Error
NAS	National Airspace System
NATDS	Next-generation Air Transportation Decision Systems
NN	Neural Network
OLAP	Online Analytical Processing
PE	Passenger Experience
PFP	Passenger Flow Predictions
PSO	Particle Swarm Optimisation
ReLU	Rectifier Linear Unit
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristics
SSDCA	Social Ski Driver Conditional Autoregressive-based
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
TSA	Transport Security and Administration
UK	United Kingdom
US	United States
WITI	Weather Impacted Traffic Index

Chapter 1: General Introduction

Flight delays are now a major problem for the global aviation industry. The air transportation industry has recently experienced rapid growth due to the increase in passengers who depend on it for their short and long-distance journeys. This high demand for air transport and the limited resources have posed a real challenge to airline and airport management. It has become more necessary and important for airlines and airport managers to manage flight schedules efficiently. For example, in 2012, the United States (US) Bureau of Transportation Statistics (BTS) data shows that 14.69% of the US domestic market flight were delayed. There was an increase to 20.8% in 2019 [1]. In 2010, Ball et al. [2] reported that in 2007 US economy was affected by as much as \$39.2 billion because of flight delays. Such a delay issue also exists in developing airline markets that are seeing steady growth, particularly in China. China, the second-largest airline market in the world, has maintained double-digit yearly growth in air traffic for the previous two decades [3]. The flight delay is a serious problem because so much expansion has been focused on urban areas, and the airport capacities have not been installed appropriately [4–6]. Flight on-time performance (FOTP) in 2019 was only 75%, with an average of 29 minutes of flight delays [3]. This situation has brought dissatisfaction to air passengers. It has also caused airlines financial losses. For example, in 1999, researchers have shown that over \$3.2 billion was spent to mitigate 22.5 million minutes of flight delay in the US. Compare this to the net profit of roughly \$7.85 billion for all airlines. It means 27% of the total net profit is a drain on financial resources [7–11]. Flight delay problem impact is tremendous to all aviation participants and has therefore necessitated major stakeholders to participate in mitigating the negative impact on the entire air transportation system (ATS) [12].

Airport congestion has been identified as one of the major causes of flight delays in China airports which may be similar to another major market [13]. Other factors such as 12% of air traffic control, 26% of extreme weather, 15% of technical and equipment failures, 11% of passengers and baggage, 8% of crew and flight operations, 8% of aircraft and ramp handling, 10% of en route and 10% related to others can also contribute to flight delays [12]. With effective

operational and managerial measures, airlines and airport operators can effectively control flight delays impact. For example, adding a “flight buffer” reduces the chances that an individual flight is delayed, and delay propagation is addressed by the flight and ground buffers, where a late arrival is a result of previous inbound flights that leads to late departure subsequent flight [14–20]. According to [13], in 2017, on average, Chinese airlines added 30.9 minutes of buffer per flight. An average buffer can exceed 40 minutes at congestion hub airports such as Beijing. Increasing buffer has been identified as a noneconomical approach as it also lowers the utilisation of aircraft, increases airlines' operational costs and wastes passengers' time which may cause dissatisfaction [21].

1.1 Motivation for the Study

This research is motivated by the high increase in the demand for air transportation over the last decades. On average, 65.4 million passengers flew in and out of the United Kingdom (UK) on 602,054 flights in 2021, with a recorded on-time flight performance of 84.00% and an average delay of 9 minutes [22]. In contrast, US domestic flights on US airlines carried 39.2 million passengers in March 2021, more than February with 14.8 million passengers. It recorded an on-time performance of 81.75%. Between 2021 and 2022, the total number of commercial flights operating in a year increased by 728,518, while the total number of time delayed in minutes increased by 3.63% [1]. Although there has been a decline in flight numbers in 2013, the total delay minutes yearly to flights has increased. The Federal Aviation Agency (FAA) forecasts an increase of 35% in flight operations in the next 20 years, i.e. each year around 1.5% increase for all commercial airlines [23], which may further increase the number of flight delays. The National Centre for Excellence for Aviation Operation Research (NEXTOR) reported in 2010 that the US domestic airlines' direct and indirect costs were around \$32 billion in 2007 [2, 24–27]. Between 2007 and 2017, the cost to airlines decreased from \$82.2 to \$68.48 per minute, but passenger costs increased from \$37.6 to \$49 per hour [2, 28, 29]. The number of commercial aircraft passengers is anticipated to increase over the following 20 years, rising from 840.8 Million in 2017 to 1.28 Billion [1]. Table 1-1 and Table 1-2 show the

summary of the US on-time performance for arrivals and departures flights from 2013 to 2022 [1].

The number of passengers is projected to increase globally per minute, substantially increasing the probability of total cost due to flight delays. Many statistical and Machine Learning approaches have been proposed to develop models that can help understand the knowledge contained in the historical flight data and predict delays. However, as aviation keeps increasing rapidly, this approach tends to be limited to the data employed in their development. Thus, this project develops state-of-the-art techniques for classifying and predicting airline flight delays with high accuracy and minimal errors. It proposes advanced Artificial Intelligence approaches such as Machine Learning and Deep Learning integrated with optimisation algorithms to generalise the models to other data sources and problems not considered in the current model development. The models for classification and prediction can assist airlines in mitigating the cost of flight delays through improved and informed decision-making.

Table 1-1 Summary of US departures on-time performance from 2013 to 2022

Year	Ontime Departures	Ontime (%)	Departure Delays	Departure Delay (%)	Cancelled (%)	Flight Operations
2013	4,683,457	80.02	1,088,021	18.59	1.39	5,852,743
2014	4,134,838	77.39	1,087,541	20.36	2.25	5,342,628
2015	4,300,421	80.53	957,607	17.93	1.53	5,339,849
2016	4,247,503	82.37	850,809	16.50	1.13	5,156,709
2017	4,206,302	80.73	926,745	17.79	1.48	5,210,416
2018	4,589,737	80.35	1,041,028	18.23	1.42	5,711,852
2019	5,435,575	79.98	1,231,567	18.12	1.90	6,796,274
2020	3,675,886	85.15	363,927	8.43	6.42	4,316,997
2021	4,453,905	81.82	899,876	16.53	1.65	5,443,512
2022	4,766,703	77.23	1,254,623	20.33	2.44	6,172,030

Table 1-2 Summary of US arrivals on-time performance from 2013 to 2022

Year	Ontime arrivals	Ontime (%)	Arrivals Delays	Arrivals Delay (%)	Cancelled (%)	Flight Operations
2013	4,634,155	79.18	1,124,319	19.21	1.39	5,852,743
2014	4,078,613	76.34	1,130,365	21.16	2.25	5,342,628
2015	4,277,624	80.11	966,667	18.10	1.53	5,339,849
2016	4,225,270	81.94	860,646	16.69	1.13	5,156,709
2017	4,177,334	80.17	944,172	18.12	1.48	5,210,416
2018	4,546,096	79.59	1,070,749	18.75	1.42	5,711,852
2019	5,387,312	79.27	1,262,308	18.57	1.90	6,796,274
2020	3,644,247	84.42	388,389	9.00	6.42	4,316,997
2021	4,449,855	81.75	891,182	16.37	1.65	5,443,512
2022	4,772,406	77.32	1,234,391	20.00	2.44	6,172,030

1.2 Aim and Objectives

To address the research gaps mentioned in this thesis, the overall scientific aim and objectives are defined as follows:

1.2.1 Aim

This research aims to develop data-driven novel and Machine Learning techniques for flight delay prediction.

1.2.2 Objectives

The aim of this research is achieved by fulfilling the stated objectives:

- i. To review previous studies and investigate the existing methods and approaches for flight delay prediction.
- ii. To identify appropriate dataset, perform pre-processing to understand and quantify the complexities, and distribution.
- iii. To identify the different data features, relations and impacts on the flight delay.
- iv. To design, implement, evaluate and validate different architectures of Machine Learning methods for flight delay prediction.
- v. To recommend the most appropriate Machine Learning architecture implementation and approach for flight delay prediction based on our experiment.

1.3 List of Publications

This project follows the scientific procedure of experimenting and producing results. Part of this thesis has been disseminated to the wider academic community through peer reviewed journal publications and national and international conferences. The list and details of all the published and unpublished outcomes of this project are summarised in Table 1-3.

Table1-3 List of publications

S/No	Manuscript Title	Journal/Conference	Status
1	A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective	5th International Conference on Advances in Artificial Intelligence (ICAAI)	Published
2	An enhanced deep autoencoder for flight delay prediction	Computational Intelligence and Neuroscience	Submitted – Under Review
3	A hybrid ensemble Machine Learning approach for arrival flight delay classification prediction using voting aggregation technique	2023 AIAA Aviation Forum: Revolutionary Leaps Toward a New Age of Aviation	Published
4	A deep bi-lstm Machine Learning method for flight delay prediction classification	Journal Of Aviation/Aerospace Education & Research	Published
5	A bidirectional deep LSTM Machine Learning method for flight delay modelling and analysis	National Training Aircraft Symposium (NTAS) 2022 - Bridging the Gap, Daytona Beach	Published
6	Social ski driver conditional autoregressive-based Deep Learning classifier for flight delay prediction	Neural Computing and Applications	Published
7	Prediction of flight delay using deep operator network with gradient-mayfly optimisation algorithm	Expert Systems With Applications	To be submitted

1.4 General Research Methodology and Thesis Structure

Figure 1-1 is an overview of the general research methodology employed in this thesis, and it is divided into four phases as follows: (i) conceptualisation, (ii) Data preparation, (iii) building a Machine Learning predictive model, (iv) evaluation and validation, and reporting the outcome in the thesis.

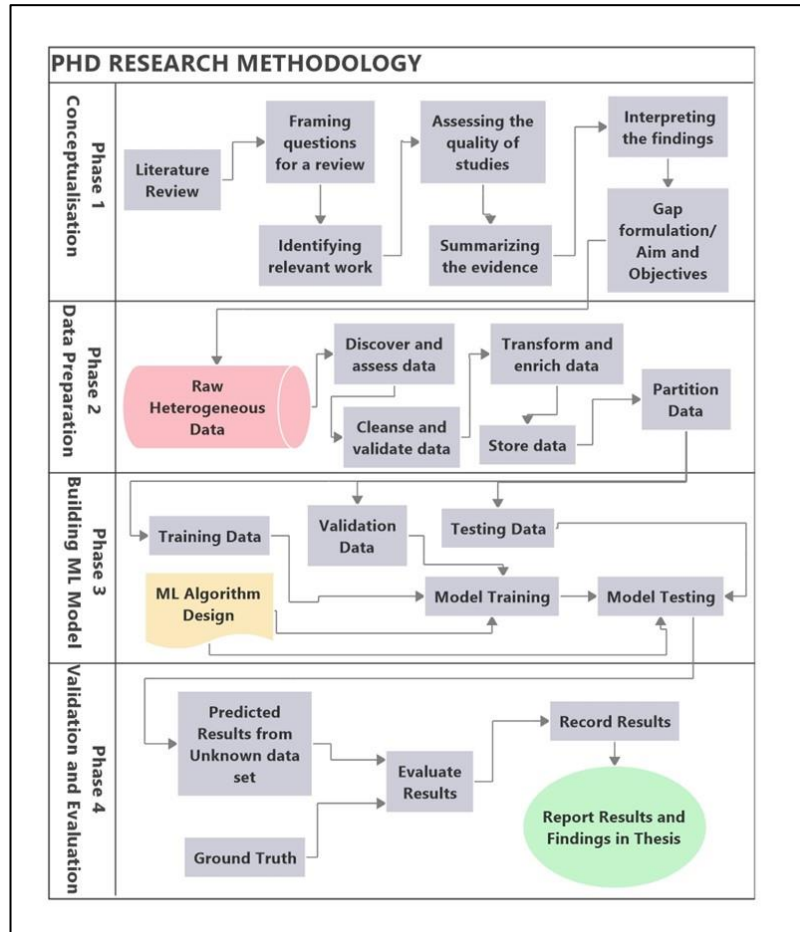


Figure 1-1 An Overview of the General PhD Research Methodology

Phase 1: This research phase involves conducting a systematic literature review, literature synthesis, gap formulation, and developing an overall aim and individual objectives guiding the study.

Phase 2: This is the stage in the Machine Learning pipeline that integrates data from various data feature sources. Data are mostly in different formats; cleaning data involves not only filling in missing values. The following operations are also needed: (i) outlier handling, (ii) inconsistency check throughout the datasets, and

(iii) selecting relevant subsets to the problem at hand. Digging deep into the data to find reasons for the different behaviour of the data involves: (i) duplicate records removal, (ii) column renaming for easy interpretability, (iii) missing values handling and (iv) proper format saving of scraped data. In contrast, analysis and visualisation involve asking relevant questions concerning business problems and values. Finally, feature engineering is important because the good feature of a simple model will always outperform the complex models with bad feature combinations. These processes are important because of their impact on the final results of the model.

Phase 3: Machine Learning and Deep Learning (ML/DL) methods have received much attention in building predictive models for flight delays over the last decades. It involves the following major stages: data acquisition, preparation, built model and evaluation.

Phase 4: This stage is where the model performance is evaluated before making recommendations.

1.5 Thesis Presentation

This thesis follows the paper format as outlined in Section 7.1 of the Cranfield University student handbook. It comprises of six technical chapters, namely Chapters 4 to 9, which are presented as self-contained concise reports designed in the format of publications. Also, the prediction of flight delays in air transport is treated as a binary classification problem. Machine Learning are used to generate a classification models (predict volume of delayed flights) and its performance is tested on real flight on-time data which is evaluated using a range of metrics. The choices of the methodologies designed and the Machine Learning methods are based on the type of dataset and the literature review conducted. This thesis also makes a significant computer science contribution to the field of aviation by proposing novel methodologies for flight delay prediction. By addressing a crucial problem in the industry, the research introduces new techniques from computer science, such as Machine Learning, Deep Learning or Artificial Intelligence, to analyse and model flight delay data. Through extensive experimentation and evaluation, the thesis showcases the superiority of the proposed methodologies

over existing approaches, offering a fresh perspective on accurately predicting flight delays. This contribution has the potential to greatly impact the aviation industry by enabling more informed decision-making, optimising flight planning and operations, and ultimately improving the overall efficiency and passenger experience through a reliable flight delays prediction.

This thesis utilises BTS dataset from the years 2000, 2013, 2018, 2019, 2020 and 2021, respectfully. This dataset is chosen for this thesis because it is open source and have been used for flight delay prediction research by other experts in the research domain [1-28]. In all the experiment the features selected for the model training and testing are based on one of the following approaches: (i) Expert Knowledge: Domain experts, such as aviation professionals or researchers, provide insights into the relevant features that impact flight delays. Their expertise helps identify key variables, such as arrival time, schedule arrival time, departure time, schedule departure time, airline, airport, historical delay patterns, and airline-specific factors. (ii) Correlation Matrix: A correlation matrix identifies the pairwise correlation between different features. It helps in identifying redundant or highly correlated features, allowing for the removal of one of them to reduce multicollinearity and improve model interpretability, or (iii) Feature Importance Techniques: These techniques, such as information gain, mutual information, or Gini index, evaluate the importance of each feature by measuring its contribution to the predictive power of the model. Features with higher importance scores are retained for modelling.

This thesis have a significant real-world impact in academia and industry. In academia, these models contribute to the advancement of research by enabling the development of new algorithms and methodologies, leading to publications and citations. In the industry, these models enhance operational efficiency by optimising flight schedules, resource allocation, and staffing levels, resulting in cost savings and improved passenger experience. They also support decision-making, contingency planning and maintenance scheduling, leading to better strategic and operational decisions. Overall, the flight delay prediction models can play a crucial role in minimising disruptions, improving productivity and enhancing the general efficiency of the aviation industry.

There are eleven chapters in the thesis and they are briefly described as follows:

Chapter 1: This chapter provides an introduction, background context and justification for the selected features used in training and validating all the proposed methodologies in this thesis.

Chapter 2: This chapter presents a systematic literature review of existing approaches and identifies potential gaps to be addressed in this thesis.

Chapter 3: This chapter explores the flight on-time dataset through exploratory data analysis, including visualisation, feature relationships and identification of hidden patterns.

Chapter 4: This chapter evaluates the classification efficacy of one hidden layer neural network, support vector machine and deep feedforward neural network with different parameter settings. Accuracy, precision, and recall metrics are used to assess the models using the US on-time dataset from BTS. The results emphasize the importance of deep neural networks for handling large datasets, but they also highlight the crucial role of pre-processing in Deep Learning and Machine Learning tasks. Diverse performance based on various parameter settings emphasises the need to find optimal configurations for reliable prediction results. The deep feedforward neural network outperforms other methods with a 20% improvement, achieving 90.01% accuracy, 90.70% precision, and 90.03% recall. It is important to note that these results do not imply equal performance across all domains, underscoring the significance of fine-tuning and feature extraction for overall model outcomes.

Chapter 5: This chapter introduces an enhanced autoencoder for flight delay prediction. The deep stack autoencoder architecture is modified to include vanilla, logistic regression and multilayer perceptron autoencoder. Through extensive evaluation on various epochs, it is observed that the deep vanilla autoencoder achieves high accuracy of 80%, with precision at 52% and recall at 98% compared to the other architectures. The confusion matrix analysis reveals that the deep vanilla autoencoder demonstrates minimal error in classifying flight

delays in the true positive and negative classes. Overall, this study highlights the significance of different architecture implementations in Machine Learning and Deep Learning approaches for flight delay prediction.

Chapter 6: This chapter introduces a hybrid ensemble Machine Learning approach for flight delay prediction using voting aggregation techniques. The ensemble model combines Logistic Regression, Decision Tree, Support Vector Classifier and Random Forest algorithms. The experiment utilises US BTS data from 2013, and the performance of the methods is evaluated using accuracy, recall, precision, F1-score, AUC and ROC curve. The extensive evaluation demonstrates that combining multiple ensemble learners improves the model's performance and learning capabilities. The proposed hybrid model surpasses individual learners in all metrics, yielding highly accurate prediction results.

Chapter 7: This chapter presents a bidirectional Long Short-Term Memory (LSTM) model for flight delay prediction, comparing it with a unidirectional LSTM. The experiment results show that the bidirectional LSTM outperforms the unidirectional one, achieving a high accuracy of 97.56% and a Mathews Correlation Coefficient (MCC) of 99%, indicating strong coherence between predicted and ground truth classes.

Chapter 8: This chapter introduces a novel Deep Learning approach for flight delay prediction. The proposed method, called SSDCA-based Deep Learning, combines the Social Ski Driver algorithm with Conditional Autoregressive Value at Risk by Regression Quantiles to optimise the learning process and enhance flight delay prediction. The Yeo-Johnson transformation is applied for data transformation. Extensive experiments are conducted to evaluate the performance of the proposed model against benchmark approaches. The results demonstrate that the proposed approach exhibits minimal error in predicting the total number of delays compared to other approaches.

Chapter 9: This chapter introduces a novel approach that enhances data pre-processing and model training using a state-of-the-art DeepONet approach integrated with optimisation algorithms. The data transformation is performed using Box-Cox, and feature fusion is achieved through a deep residual network.

Prediction is carried out using the Gradient Mayfly Optimisation Algorithm (GMOA). The experiment results emphasise the significance of optimisation in developing reliable flight prediction models with minimal error rates and generalisability.

Chapter 10: This chapter discusses the overall findings of the thesis, emphasising their real-world impact in academia and industry.

Chapter 11: This chapter concludes the thesis and outlines future work for extension, including exploring other real-world application areas.

1.6 References

1. Bureau of Transportation Statistics (BTS). Airline On-Time Statistics and Delay Causes: On-Time Arrival Performance. United States Department of Transportation; 2022. Available at: https://www.transtats.bts.gov/OT_Delay/ot_delaycause1.asp?type=1&pn=1%0Ahttps://www.transtats.bts.gov/OT_Delay/OT_DelayCause1.asp?pn=1%0Ahttps://www.transtats.bts.gov/OT_Delay/OT_DelayCause1.asp (Accessed: 4 March 2023).
2. Ball M., Barnhart C., Dresner M., Hansen M., Neels K., Odoni A., et al. Total delay impact study: a comprehensive assessment of the costs and impacts of flight delay in the United States. Nextor. 2010; : 5–14, 25–30. Available at: http://www.isr.umd.edu/NEXTOR/pubs/TDI_Report_Final_10_18_10_V3.pdf.
3. Civil Aviation Administration of China (CAAC). Statistics of Key Performance Indicators for China's Civil Aviation Industry in December 2022. 2022.
4. Kalakou S., Psaraki-Kalouptsidi V., Moura F. Future airport terminals: New technologies promise capacity gains. Journal of Air Transport Management. 2015; 42: 203–212. Available at: DOI:10.1016/j.jairtraman.2014.10.005.
5. Ben Messaoud M. A thorough review of aircraft landing operation from practical and theoretical standpoints at an airport which may include a single or multiple runways. Applied Soft Computing. Elsevier B.V.; 2021; 98: 106853. Available at: DOI:10.1016/j.asoc.2020.106853.
6. Industry High-level Group. Aviation Benefits Report. 2019; : 76.
7. Tang Z., Huang S., Zhu X., Pan W., Han S., Gong T. Research on the multilayer structure of flight delay in China air traffic network. Physica A: Statistical Mechanics and its Applications. Elsevier B.V.; 2023; 609: 128309. Available at: DOI:10.1016/j.physa.2022.128309.
8. PeCoy MD., Redmond MA. Flight reliability during periods of high uncertainty. Journal of Air Transport Management. Elsevier Ltd; 2023; 106(August 2022). Available at: DOI:10.1016/j.jairtraman.2022.102322.
9. Abdelghany A., Guzhva VS., Abdelghany K. The limitation of machine-learning based models in predicting airline flight block time. Journal of Air

- Transport Management. Elsevier Ltd; 2023; 107(November 2021): 102339. Available at: DOI:10.1016/j.jairtraman.2022.102339.
10. Liu C., Chen YR., Wang H., Zhang YY., Dai X., Luo Q., et al. Airport flight ground service time prediction with missing data using graph convolutional neural network imputation and bidirectional sliding mechanism. *Applied Soft Computing*. Elsevier B.V.; 2023; 133: 109941. Available at: DOI:10.1016/j.asoc.2022.109941.
 11. Fatima SK., Campus A., Abbas SM., Campus A., Mir I., Campus A., et al. A Comprehensive Framework For Model Prediction of Aerial Vehicles : Perspective Analysis and Comparison. 2023; (January).
 12. Tan X., Jia R., Yan J., Wang K., Bian L. An Exploratory analysis of flight delay propagation in China. *Journal of Air Transport Management*. Elsevier Ltd; 2021; 92(February 2020): 102025. Available at: DOI:10.1016/j.jairtraman.2021.102025.
 13. Institute for Aviation Research (IAR) China-Europe Aviation. On-Time Performance Research for Mainland China. Report. 2018.
 14. Wu Y., Mei G., Shao K. Revealing Influence of Meteorological Conditions and Flight Factors on Delays Using XGBoost. *Journal of Computational Mathematics and Data Science*. Elsevier B.V.; 2022; : 100030. Available at: DOI:10.1016/j.jcmds.2022.100030
 15. Giusti L., Carvalho L., Gomes AT., Coutinho R., Soares J., Ogasawara E. Analyzing flight delay prediction under concept drift. *Evolving Systems*. Springer Berlin Heidelberg; 2022; (0123456789). Available at: DOI:10.1007/s12530-021-09415-z.
 16. Mellenbergh G. Tests and questionnaires. *Advising on research methods: A consultant's companion*. 2008; : 211–236. Available at: [https://books.google.com/books?hl=en&lr=&id=LCnOj4ZFyjkC&oi=fnd&pg=PA13&dq=Mellenbergh,+G.J.+\(2008\).+Chapter+10:+Tests+and+Questionnaires:+Construction+and+administration.+In+H.J.+Adèr+%26+G.J.+Mellenbergh+\(Eds.\)+\(with+contributions+by+D.J.+Hand\),+Advisin](https://books.google.com/books?hl=en&lr=&id=LCnOj4ZFyjkC&oi=fnd&pg=PA13&dq=Mellenbergh,+G.J.+(2008).+Chapter+10:+Tests+and+Questionnaires:+Construction+and+administration.+In+H.J.+Adèr+%26+G.J.+Mellenbergh+(Eds.)+(with+contributions+by+D.J.+Hand),+Advisin)
 17. Lan S. Planning for robust airline operations: optimizing aircraft routings and flight departure times to achieve minimum passenger disruptions. *Massachusetts Institute of Technology*. 2003. Available at: <https://dspace.mit.edu/bitstream/handle/1721.1/17568/52769892-MIT.pdf?sequence=2&isAllowed=y>.
 18. Kockelman K., Chen TD., Street G. A Transportation Economics Reference for Practitioners: Final Report. 2013. Available at: <https://rosap.ntl.bts.gov/view/dot/25833>.
 19. Farrokhi A., Farahbakhsh R., Rezazadeh J., Minerva R. Application of Internet of Things and Artificial Intelligence for smart fitness: A survey. *Computer Networks*. Elsevier B.V.; 2021; : 107859. Available at: DOI:10.1016/j.comnet.2021.107859.
 20. Sharma N., Sharma R., Jindal N. Machine Learning and Deep Learning Applications-A Vision Neha. *Global Transitions Proceedings*. Elsevier B.V.; 2021; : 0–8. Available at: DOI:10.1016/j.gltp.2021.01.004.
 21. Qu J., Wu S., Zhang J. Flight Delay Propagation Prediction Based on Deep Learning. *Mathematics*. 2023; 11(3): 494. Available at: DOI:10.3390/math11030494.

22. CAA. 2021 Quarter Four Flight Data. UK Civil Aviation Authority Statistics. 2022. Available at: <https://www.caa.co.uk/news/2021-quarter-four-flight-data/> (Accessed: 4 March 2023).
23. United States Department of Transportation. Federal Aviation Administration. Washington: Office of Inspector General, U.S. Department of Transportation; 2020. Available at: <https://www.faa.gov/> (Accessed: 20 June 2021).
24. Ball MO., Hoffman R., Mukherjee A. Ground delay program planning under uncertainty based on the ration-by-distance principle. *Transportation Science*. 2010; 44(1): 1–14. Available at: DOI:10.1287/trsc.1090.0289.
25. Kalliguddi AM., Leboulluec AK. Predictive Modelling of Aircraft Flight Delay. *Universal Journal of Management*. 2017; 5(10): 485–491. Available at: DOI:10.13189/ujm.2017.051003.
26. Ferguson J., Kara AQ., Hoffman K., Sherry L. Estimating domestic US airline cost of delay based on European model. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2013; 33: 311–323. Available at: DOI:10.1016/j.trc.2011.10.003.
27. Dalmau R., Ballerini F., Naessens H., Belkoura S., Wangnick S. An explainable Machine Learning approach to improve take-off time predictions. *Journal of Air Transport Management*. Elsevier Ltd; 2021; 95(June): 102090. Available at: DOI:10.1016/j.jairtraman.2021.102090.
28. Ball M. O., Chen C. Y., Hoffman R. and VT. Collaborative decision making in air traffic management: Current and future research directions. In *New Concepts and Methods in Air Traffic Management*. 2001. Available at: DOI:10.1007/978-3-642-56680-6.

Chapter 2: A Systematic Literature Review on Flight Delays Prediction Methods

Passenger experience (PE) or satisfaction (PS) is crucial for businesses, including airlines and airports. With the growing number of air travelers, there is a need for solutions to enhance their experience. However, failures from both airlines and airports can lead to variations in passenger experience, causing dissatisfaction. This review focuses on three major problems faced by air transportation passengers: flight delay predictions (FDP), passenger flow predictions (PFP), and baggage handling (BH). Addressing these issues can aid stakeholders in the aviation transport system to make informed policies and guidelines, ultimately leading to passenger happiness. To tackle these problems and improve passenger experience, it is vital to develop models and solutions that provide advance information to key decision-makers and passengers due to the complexity of the aviation transport system. In addition, we conducted an extensive literature study from a data science perspective to explore state-of-the-art methods for solving these problems and building predictive models. We developed a taxonomy and summarised various approaches employed in the research. We also established a connection between the methods used and the specific passenger problems addressed over time. By reviewing the existing literature, we identified current and future research trends in this field. Finally, we provided recommendations to further enhance the studies on air passenger experience through reliable predictive modelling.

2.1 Introduction

An air passenger's experience involves how the passenger feels, think, reason, select, apply products and services related to air travel. Understanding the passenger experience (PE) and future response is essential for airlines and airports to improve their services, products, and marketing methods. Meeting the passengers' expectations will lead to long-term profit in the aviation industry [1]. In digital aviation, passenger experience has been a pertinent issue in all businesses, regardless of the nature or mode of service delivery. The satisfaction of the customers is paramount to the business owners and stakeholders.

Measuring passenger satisfaction has been a daunting issue for ages. Each passenger may have different experiences concerning different service situations, making it challenging to know the exact answers to which customer is truly satisfied or not satisfied without direct interactions with them individually [2]. According to the Airport's Council International (ACI) report in 2009, more than 50% of traffic at some large hub airports is due to service quality, a more determining factor when considering passenger connections [1].

There is an increase in the number of flights and passengers worldwide, with an average growth rate of about 3.8% by 2030, as reported by the International Airport Transport Authority (IATA). Air passenger Worldwide travel number continued to rise, exceeding 4.3 billion journeys in 2018. Therefore, efficient management of the air passenger is necessary. Because this can positively impact passenger experience and the airport's financial performance [3], reports indicate that excellent experience can influence future travel plans and encourage spending [4]. The overall financial portfolio of companies operating in the airports is mainly on the retail concession business. In some cases, 53% of total revenues are accounted for non-aviation revenues [5-6]. There have been projects going on recently in the area of airport passenger experience as follows: Proactive Passenger Flow Management for Airport with an Advanced Forecasting System (AERFOR) used for managing and predicting passenger flow within the terminal; Airport of the Future Project (AFP) this focuses on the safety, security and efficiency which is applied in the Australian airports; and the DORA project which aims at providing passengers with optimize and total travel time taking into account starting and destination point of the air passengers.

One of the most applied performance indicators in airports and airline is how well passengers experience the quality of services. A period in which a flight is late for at least 15 minutes is said to be delayed. Thus, the difference between the real-time and scheduled time of departure or arrival of the plane represents its delay [5]. Many indicators relate to tolerance peak points for delays by different country regulatory authorities. In the past decades, passenger experience flight delay has been an important research area in the air transport system. In the year 2013 number of flight delays recorded with more than five minutes was 36% in

Europe, while flight delays with more than fifteen minutes in the United States were 31.1%, and several cancelled or flights delays with more than thirty minutes in Brazil was 16.3% [6–26]. It shows the high relevance of flight delay research and its negative effect on airline meshes regardless of scale.

The rest of the paper is as follows: Section 2.1 discusses the passenger experience from a data science perspective and ways of addressing the problem of air passenger experience. Section 2.3 has the result analysis of passenger experience in a taxonomic manner presented with the authors' data and methods, application scope, and most researched passenger experience. Section 2.4 discuss the results in terms of publication timeline categorized by problem types and their point of relationship. Finally, Section 2.5 concludes the survey outputs by highlighting and presenting future passenger experience problem research using Machine Learning and data.

2.2 A Data Scientist Perspective of Air Passenger Experience

Discovering knowledge from data under acceptable computational efficiency limitations and enumerating particular patterns using some algorithms [27]. Due to the enormous amount of data in their systems, the civil aviation industry is one of the earliest adopters of data science. The aviation industry's use of analytics and intelligence of technological techniques begins at the end of the '60s. Over the years, this industry has made considerable investments to find useful patterns from their generated data and explore more improved operational efficiency methods to encourage customer loyalty. One of the crucial 21st-century sectors is the aviation sector. We consider civil and commercial aviation. It has become evident that data is a critical part of a successful data science application [11–13].

The other point of note is that aircraft record their data always in their black box. Up to 500 data variables are described in planes equipped with flight recording capabilities. These data recordings in flight are altitude, time, vertical acceleration, and heading per second for the whole operation duration [28]. Manually finding patterns and massive data is practically impossible because of the large amount of such data generated daily [29]. The need to use descriptive behaviour, a scalable analytics service by the airline companies' domain to better

the sense-making in the current and historical data for predictions, is on the high side [30]. The vast amount of data generated by the civil aviation sector' contains massive information about the passenger's experience, which could be further used to transform the industry through improved service delivery.

Due to the complexity and distributed nature of commercial aviation transport systems, data are generated from aircraft, passengers, and other activities, such as non-aviation sections of the airports. Orchestration is needed for smooth safety operations because it deals with valuable resources, fluctuations in demand, and a sophisticated origin-destination matrix. Each passenger follows their itineraries as the airlines plan various aircraft schedules, pilots, and flight destinations. Several stages are involved at airports, runways, and airspace at the terminal boundaries, which may lead a passenger to experience different types, such as flight delays or dissatisfaction due to the outcome of these processes. Examples include weather conditions, air traffic control, capacity constraints, run-away queues, and mechanical problems [29–32].

This process continues for each flight in the system throughout the day several times. Due to legal duties, rest, and aeroplanes maintenance plans, different schedules may be applied to pilots, aircraft, and flight attendants. Therefore, any inconsistency in the system can affect the same airlines' succeeding flights [24, 33–35]. Furthermore, congestion at other airports or airspace may be caused by disturbances, creating delays and queues of other carriers' flights with so much harmful passenger experience data generated for such periods [33, 36, 37]. Because of this, the prediction of delays or passenger flow and other aspects of the travel processes to improve passenger experience except for unforeseen circumstances is an essential topic for airports, airlines and network managers like Eurocontrol and Federal Aviation Administration (FAA), and Air Navigation Service Providers (ANSP) [38, 39].

Passenger experience problems can be from different perspectives, such as (i) passenger flow prediction, (ii) flight delay propagation, and (iii) root delay and cancellation [40, 41, 42–44]. In the passenger flow prediction, one studied how to optimise passengers' control from the initial check-in points until boarding into the aircraft. While in flight delay propagation, delays through the transport

network system are explained thoroughly. Also, considering that there might be new problems eventually, the causes and occurrences of further delays should be predicted and understood [45]. Such a type of occurrence is said to be the *root* delay problem. Under certain situations, cancellations may occur due to delays, causing passengers and airlines to change their itinerary schedules. Many researchers focused more on cancellation analysis, which helped them determine the conditions that lead to cancellations. Which airline's decision-making process is explored in choosing flights that will be cancelled? Figure 2-1 shows a typical flow of traditional commercial flight operations, including all the necessary processes from origin to destination airport, as proposed by [5], for a data scientist. Data generated from any process is essential to propose a solution for improving and optimising service because of the explanatory power of real-life data to infer the future from its hidden historical patterns when properly mined by a data scientist to create model solutions.

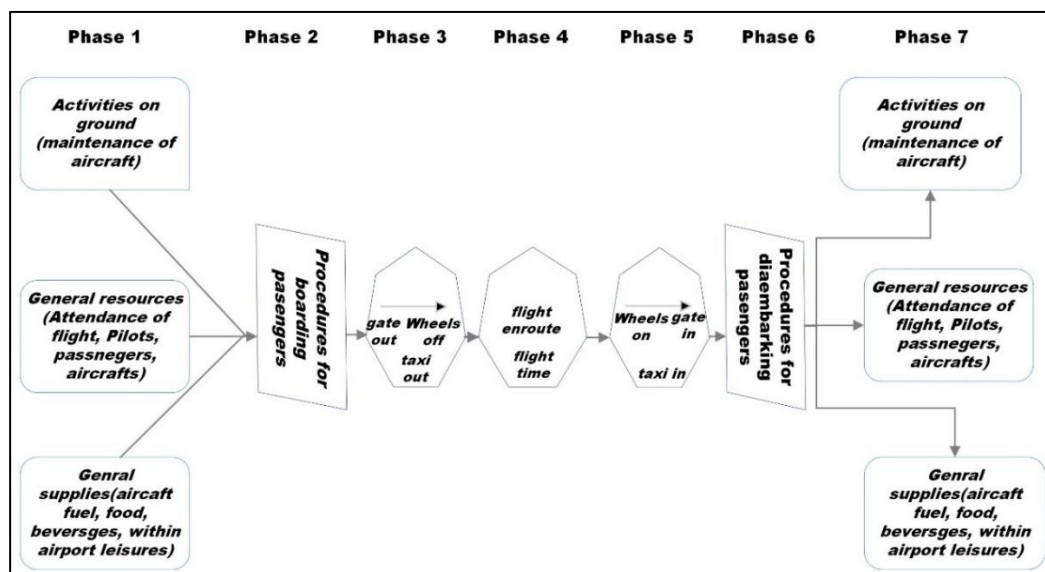


Figure 2-1 A traditional Commercial Flight Operation

2.3 A Taxonomic Analysis of Passenger Experience Problem

Several problems are involved in the passenger experience, ranging from predicting flight delays, passenger flows, passenger segmentation, passenger profiling and security. In general, passenger experience has the following related problems that can be used to build a taxonomy of passenger experience problems. These are models, scopes and procedures for handling passenger

experience problems. Figure 2-2 shows the taxonomic view of the passenger experience problem and the various subsections' explanations. A systematic mapping study was conducted regarding passenger experience from Machine Learning perspectives to search for available literature.

Several search expression strings have been designed to capture this review's entire area of interest. The expression string used for search on Scopus, Google Scholar, IEEE and ACM databases is (*“Machine Learning”* $\wedge$ *“data”* $\wedge$ *“passenger experience”*) $\vee$ (*“Machine Learning”* $\wedge$ *“data”* $\wedge$ *“flight delay”* $\wedge$ *“prediction”*) $\vee$ (*“Machine Learning”* $\wedge$ *“flight delay”* $\wedge$ *“prediction”*) $\vee$ (*“data”* $\wedge$ *“flight delay”* $\wedge$ *“prediction”*) $\vee$ (*“Machine Learning”* $\wedge$ *“data”* $\wedge$ *“airport passenger flow”* $\wedge$ *“prediction”*) $\vee$ (*“Machine Learning”* $\wedge$ *“airport passenger flow”* $\wedge$ *“prediction”*) $\vee$ (*“data”* $\wedge$ *“airport passenger flow”* $\wedge$ *“prediction”*) from 1997 to April 2023, respectively. The snowballing approach for this review work was proposed by Wohlin in 2014, as shown in Figure 2-3 [46]. The results extracted from the search expression string query on Scopus was one hundred and eighty-six, and the snowball search technique brought three hundred more articles. We selected two hundred papers explicitly for the analysis from the snowballing method. Some of the articles are duplicated and thus discarded. After selecting the most relevant articles, the query results from Scopus, Google Scholar, IEEE and ACM databases are shown in Table 2-1, containing the summary of the number of articles extracted per year from the search expression string and snowballing search used in this paper for analysis purposes. We set a criterion for the relevant article's inclusion in the review. It must include either one or the combination of the following words: *“Machine Learning”*, *“data”*, *“flight”*, *“delay”*, *“passenger”*, and *“prediction”* in the abstract, having at least one Google Scholar citation per year to April 2023 inclusive.

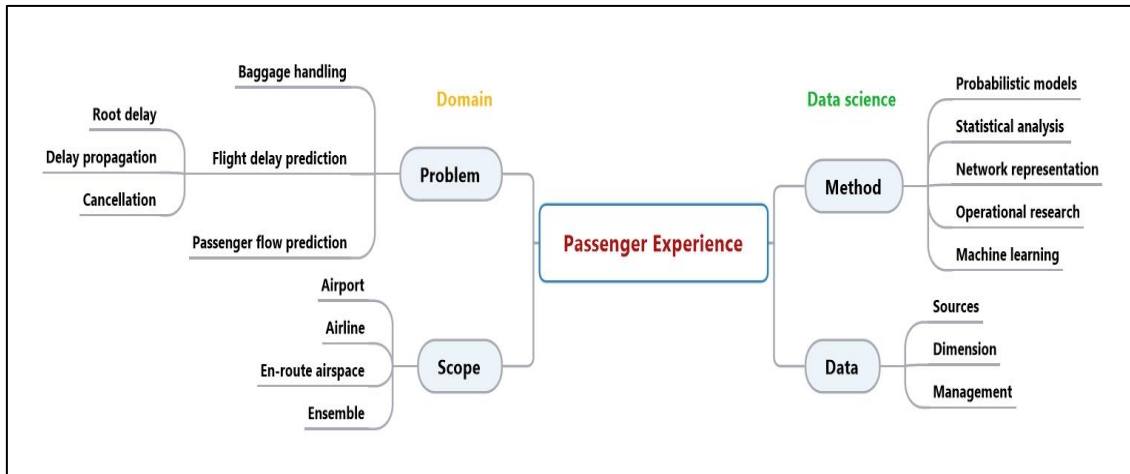


Figure 2-2 Taxonomy of Passenger Experience Problem

The taxonomy's components are discussed in Section 2.3.1 about this review paper's targeted objective, passenger experience.

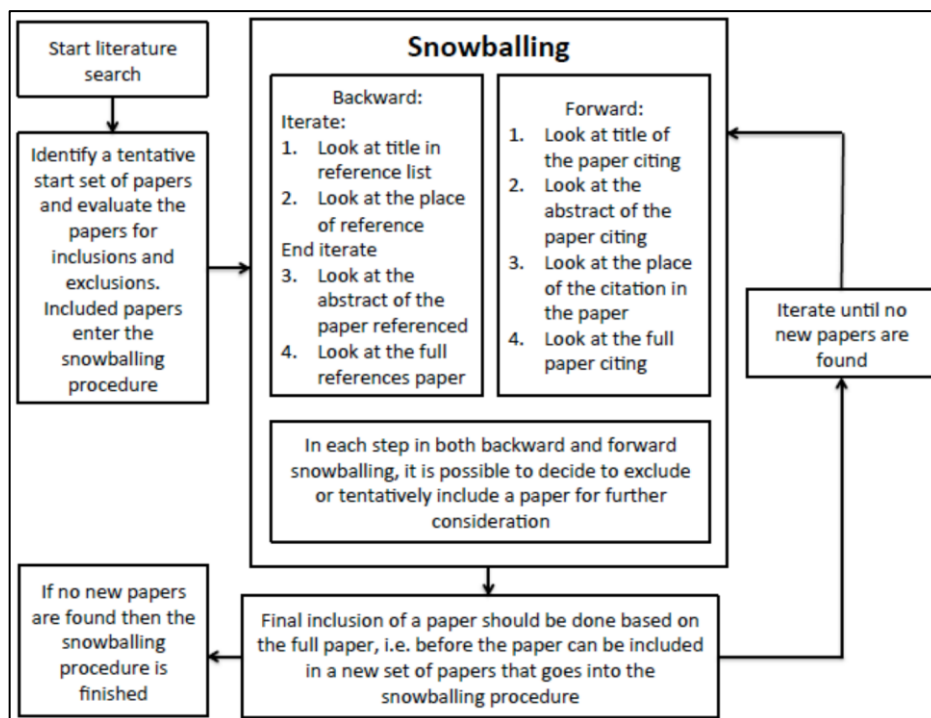


Figure 2-3 Procedure for Snowballing Search [46]

For traceability purposes, only one iteration is performed for backwards and forward snowballing to identify new papers included in the pile of papers that will go to the next iteration phase. We repeated the process until the end of the review with the process in Figure 2-3.

Table 2-1 A summary result of extracted papers for analysis

S/No.	Search string	Year	Number of articles extracted
1	(<i>“Machine Learning”</i> ^ <i>“data”</i> ^ <i>“passenger experience”</i>)	2020	01
2	(“Machine Learning” ^ “data” ^ “flight delay” ^ “prediction”)	2016	02
		2017	01
		2018	01
		2019	04
		2020	01
3	(“Machine Learning” ^ “flight delay” ^ “prediction”)	2010	01
		2011	02
		2012	02
		2013	01
		2015	02
		2016	04
		2017	02
		2018	08
		2019	18
		2020	02
4	(“data” ^ “flight delay” ^ “prediction”)	2012	01
		2015	01
		2016	04
		2017	03
		2019	05
	(“Machine Learning” ^ “data” ^ “airport passenger flow” ^ “prediction”)	2020	01
5		2017	02
		2019	02
6	(“Machine Learning” ^ “airport passenger flow” ^ “prediction”)	2020	01
		2017	01
7	(“data” ^ “airport passenger flow” ^ “prediction”)	2018	01
8	(“data” ^ “flight delay” ^ “prediction”)	2017	01
8	(“data” ^ “flight delay” ^ “prediction”)	2021	03
9	(“data” ^ “flight delay” ^ “prediction”)	2022	08
10	(“data” ^ “flight delay” ^ “prediction”)	2023	02
8	Snowballing search technique		160
Total Extracted Articles			200

The snowballing search procedure summary is shown in Figure 2-3. It has steps such as the starter set, which looks at the initial set of keywords and search expression strings that will be applied; the start set of papers will be included in the major challenge. At the snowballing stage, there are two significant steps which are the backward and forward iterations. The backward snowballing

involves using a reference list to identify which new paper can be added to the original list. It is done by scanning the already-used list of papers based on criteria such as language, year of publication, and type of publication. The next step of the snowballing stage is the forward snowballing stage has to do with choosing papers based on the types of papers that cite them. Google Scholar citations are being used to examine the citation level of the inclusion papers. After this is done, the paper's abstract is read to check for keywords of interest; if this is satisfied, then the next stage of inclusion or exclusion is performed. At the stage of inclusion or exclusion, the target is to include the right papers. If any paper included is discovered not relevant, it is then excluded. Only papers that satisfy the inclusion criteria are used for the paper's analysis aspect.

2.3.1 Problem

The taxonomy is formulated with four components. The components, the problem, are the core because they form the foundations for finding solutions in any field. From Figure 2-2, there is three major passenger experience problem formulated in this research which is as follows: baggage handling, flight delay prediction and passenger flow prediction or crowd control [47]. Based on the research's primary focus, the authors try to solve any three passenger experience problems by developing models in line with the set objectives.

Baggage handling

Baggage or luggage handling is one of the significant contributors to the quality of the airline or airport. Because of the constant competition increase in the air transport industry, there is a constant need to improve baggage handling, causing severe pressure on the airport/airline companies. The baggage handling system is highly technical; this may cause the temptation to focus only on the airport management's technical aspect [48]. Despite all the approaches given to the system being designed to handle the peak flow bottleneck, it may make the system large and unnecessarily complicated. Therefore, researchers try to propose a process view that considers the entire process chain's optimisation to prevent the bottlenecks' occurrence as much as possible. Several parties involved in baggage handling cause some bottlenecks because of their peculiar demands on the process. The parties are Passengers, airlines, airports, security,

and handling agents [49]. This challenge is to conduct more research and develop more technologies to help handle the problem in the future to improve the passenger's experience by minimising the bottleneck caused by different parties in the process as much as possible.

Flight delay and propagation prediction

Flight delay and its propagation prediction is a huge topic of interest that has received much attention from research over passenger experience improvement research. Airlines and industry partners have invested lots of funds to build Machine Learning models to reduce the bottleneck as minimally as possible. Some research done in this area is [5, 11, 56–65, 34, 30, 35, 50–55, 66–71, 79, 80–82, 90, 91, 92]. Based on this review, we have identified three major problems that researchers have been trying to improve over the years. They are as follows: root delay, delay propagation, and cancellation, as indicated in Figure 2-2.

Selected features for model training

In the field of Machine Learning, the selection of appropriate features plays a crucial role in model training and performance. It is a process that requires careful consideration and expert analysis to identify the most relevant features for accurate predictions. This is true both for the research conducted in this thesis and the expert feature selection observed in the extant literature.

Table 2-2 serves as a comprehensive overview of the selected features for training machine learning models in this thesis, alongside the expert feature selection found in the extant literature. It provides a valuable reference point for understanding the significance and impact of feature selection on model outcomes.

By examining the selected features in Table 2-2, insights can be gained into the variables and attributes that have been deemed essential for successful model training. This comparative analysis allows us to identify commonalities, patterns and areas of divergence, providing valuable guidance for future research in the field of flight delay predictive modelling.

Table 2-2 Selected input features for this thesis and other expert opinion selected features in literature

REF.	T	Y	M	DOM	DOW	DT	SDT	AT	SAT	UC	FN	TN	AET	SET	AD	DD	O	D	DT	TI	TO	DI	CD	WD	NASD	SD	LAD	OTHERS
[81]			X	X	X												X						X				X	
[80]					X	X	X	X	X			X		X	X	X	X	X			X							X
[79]						X	X	X	X				X	X	X	X	X	X	X		X							
[78]				X	X	X	X	X	X	X							X	X	X	X				X				
[77]		X	X	X	X					X	X	X			X		X	X	X	X					X			X
[76]		X	X	X			X			X				X		X	X	X	X	X								
[44]		X		X	X	X	X	X				X				X	X	X	X								X	
[15]					X	X	X	X	X	X	X		X				X	X					X	X	X			
[75]						X	X	X	X	X		X					X	X									X	
[74]				X	X	X	X	X	X							X	X				X		X	X	X			
[177]						X	X	X	X	X		X					X	X										
[247]		X	X	X	X	X		X			X				X	X	X	X					X	X	X	X	X	
[73]				X	X	X	X	X	X						X	X	X	X										
[205]		X	X	X	X	X	X	X	X	X		X	X	X			X	X	X	X				X				
[72]						X	X	X	X	X		X					X	X	X					X				X
[18]						X	X	X	X	X	X	X	X	X	X	X	X	X										X
T			X	X		X	X	X	X	X		X	X	X	X	X	X	X										
REF.		Y	M	DOM	DOW	DT	SDT	AT	SAT	UC	FN	TN	AET	SET	AD	DD	O	D	DT	TI	TO	DI	CD	WD	NASD	SD	LAD	OTHERS

travelling. Aircraft as essential transportation means that experience of massive data from traffic increases in various airports. An airline network links large and medium cities because it is an essential transportation means. Passenger traffic in airlines and airports are vital resource [93]. Air resource configuration can improve when the airlines' temporal and spatial distribution is adequately analysed [94]. When passengers are rationally allocated into the proper flow, such as departure flow, it will helpfully use the airport system resources and optimise the flight flow.

Accurate prediction of passenger flow using historical data will be essential to improving the passenger experience. It will also help in passenger flow planning decisions. Some researchers have tried to analyse passenger flow factors at different locations and times through network data and questionnaires [95–97]. However, research on airport passenger flow has received little attention, unlike the research on bus park passenger flow, which has received much attention [98–100], to mention but a few. Therefore this indicates the gap that needs more investigation now and in the future [101].

2.3.2 Scope

A different air transport system source can cause passenger experience problems affecting airports, *en route* airspace or an ensemble, and airports. For analysis, only one or any combination of the mentioned actors may be considered based on having a simplified system. The problems discussed in Section 2.3.1 may be combined with any application scope in building a model that can help improve passenger experience problems [5].

Predicting delays models by some researchers focus on all *en-route* airspace and airlines indifferently for all the departs delay [102, 103]. When the main target is investigating efficiency based on all carrier delays in airports and being focused upon [104–108], only the airlines are considered when comparing two airlines based on the same conditions [109].

The relationship between delays and congestion was studied and understood through ensemble *en-route* and airport [110–112]. In contrast, others consider airlines and airports to evaluate airlines' capacity and decisions [113]. Ensemble

scopes can have many possibilities. When targeting root delay, studying the dynamics of transportation systems becomes essential.

2.3.3 Data

There is an inevitable question about every piece of data that is generated for improving passenger experience problem considering the five V's characteristics of Data, which are: volume, this has to do with the various sizes of the data ranging from Terabytes, Exabytes, and Zettabytes and beyond, veracity, this has to do with the various uncertainty, incompleteness, and inconsistency of the data; velocity, this has to do with the rate and method in which the data is generated such as is it in batches, real-time, or streaming; variety, this deal with the nature of data looking at properties such as structure or unstructured; and value, this looks at what the final result will help gain from the data analysis such data into money or data into actions. They are: Where can we find the most appropriate data? Which attributes of the data should be considered? Can each datum be a source for better results? Answering these questions can be related to the three passenger experience data branches, as shown in Figure 2-4. These are (i) sources, (ii) dimension, and (iii) management.

Sources of Data

Airport passenger experience data from the air transportation systems are related to airports or ensembles, airlines, and passengers. Databases of airports and airlines are not shared among the entire community; only those institutions' collaborators use them. Ensemble datasets sometimes include airports, carriers, regulatory authorities, service providers, and additional government agency information. Table 2-2 lists some datasets in the regional order. Many publications are presented using the datasets, and the topmost cited are indicated in each category. Government agencies provide different granularity of public access databases. It is worth stating that the most widely used datasets to obtain information about flights are provided by The US Department of Transportation through The Bureau of Transportation Statistics and The Federal Aviation Administration (<https://www.transportation.gov>; <https://www.bts.gov/>; <https://www.faa.gov>). The United Kingdom Civil Aviation is another primary

source of datasets for the air transport system supported by the London Datastore and Office for National Survey (<https://www.caa.co.uk>; <https://www.ons.gov.uk>; <https://data.london.gov.uk/>). Eurocontrol provided the Europe intergovernmental organisation database (<https://www.faa.gov>). It is also more focused on studies dealing with passenger experience through flight delay studies.

Other related datasets exist to handle other factors that help build the passenger experience models, such as weather, which may be provided by private service providers' databases or government databases. For example, the US National Oceanic and Atmospheric Administration, the US Department of Transportation Statistics and the UK Met Office. Weather Companies are used to develop some models by researchers from different sources. (<https://www.noaa.gov/>; <https://www.transportation.gov/>; <https://www.metoffice.gov.uk/services/data>).

Synthetic datasets are created by some researchers for their models' evaluation instead of the use of datasets provided by these agencies responsible for real-life such datasets, for example, a scenario of the market with consideration of link, airport capacity, frequency, passenger demand, and flight characteristics by Zou et al. [149–151].

Table 2-3 Air transportation system real data sources per region

S/No.	Region	Airline	Airport	Ensemble
1	Asia	01 [26, 71, 120, 121]	01 [122, 123]	02 [124,125]
2	Brazil	N/A	N/A	02 [5, 126, 53]
3	Europe	02 [127, 127]	07 [128, 129, 130, 131]	07 [132, 134, 85]
4	United Kingdom	05 [85, 135, 136]	11 [135, 136, 137]	07 [85, 105, 138]
5	United States	07 [139, 140, 141]	16 [40, 142, 143, 144]	11 [113, 145, 146, 147]

Dimensions

Based on the papers we analysed and the available public datasets, we organised seven main standard classes of attributes used to build models for passenger experience Improvement in the past, as shown in Figure 2-4. These

attributes are an abstraction of the input attributes mainly applied in building models for prediction in any passenger experience improvement research related to flight delay prediction, passenger flow prediction or baggage handling, which are the focus of this review article. Several characteristics depend on the research's focus beyond arrival and departure times.

The positions taken by an aircraft, such as arrival and departure airports, regions, countries, and cities, are referred to as its spatial dimension [152–154]. Data's seasonality and periodic pattern are often seen as its temporal dimension. Time (Time of the day or the day) and date (Day of the week, month, and season) are both characteristics of these elements [37, 54, 56, 79, 81, 155]. The weather explains the external and environmental conditions at a particular time [110]. Specific features such as visibility and ceiling may be represented by defining instrumental or visual conditions during take-off or landing [128]. Several short-term parameters are contained in airport weather situations (i.e. surface weather) and *en-route* weather situations (i.e. convective weather) [101, 156–158].

In baggage handling, the detail of the passenger who owns the baggage and the total number of baggage that must not exceed the maximum acceptable weight allowed per passenger is indicated. The record of the passenger indicates the passenger's socio-demographics and trip details. Other retail detail details are kept based on the total non-aviation income amount within a specific window under consideration. While in planning, critical resource involves the operations of airports, airlines, and traffic controllers. Flight plans, airport schedules, and airport schedules are included in this dimension. The airline schedule defines the origin and destination points, their sequence and frequency, and aircraft and flight crew allocations [50, 71, 144, 159–163]. The airport schedules indicate the take-off and landing each time, while *en-route* parameters dimensions such as route, speed and height, and distance indicate the flight. The features represent characteristics of aircraft or airports and airlines. Major, affiliate low-cost point-to-point or traditional hub-and-spoke are represented by features. The size, number of seats, and aircraft occupancy may affect market decisions and constrain some operations. The number of runways and service providers is represented by airport infrastructure in the air facility, as shown in Figure 2-4.

The specific operating conditions in which the system's state indicates airports or en route, airlines, are now. Some prior levels of delay or airport closure of airport examples correspond to the research by Zonglei et al. [164]. The information of the system predicts the behaviour of the system. Lastly, *en-route* and airports, demand and capacity are related operations. A congestion scenario could be formed when demand exceeds capacity, which causes delays [101, 165, 166].

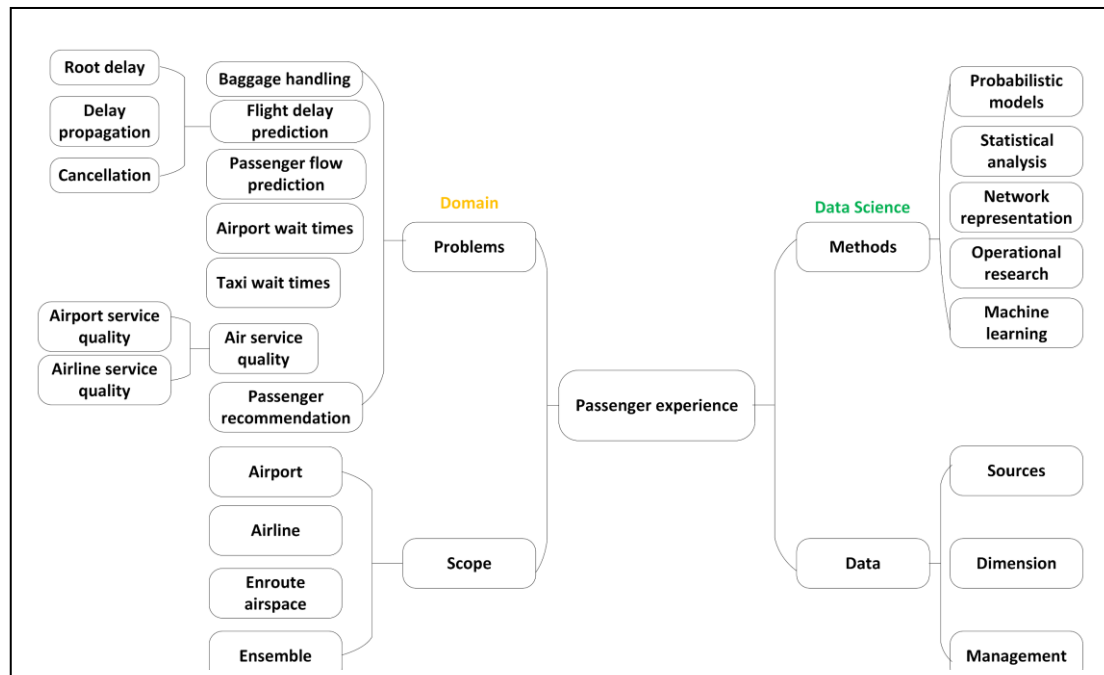


Figure 2-4 Passenger Experience Data Model

Management

Over the last decades, databases to store the enormous data generated by various sections of human activities have increased. Therefore, this has made managing this data more crucial for efficiently and conveniently handling queries. The database system design enabled a structural way of integrating data from different sources, transforming the data, and eliminating inconsistencies. For this purpose, developing a data warehouse that supports managing data and online analytical processing (OLAP) will be useful. In Section 3.3.1, it was stated that different sources of data might be employed. Therefore, to link these different data sources, using a data warehouse with the Extract, Transform and Load (ETL) methods are needed for better and optimised solutions [5, 167].

Several data pre-processing techniques can be applied to generate datasets from passengers, flights, and aircraft. Removing outliers from data is one of the major tasks in data cleaning. Anomalies may be caused in the model results by outliers from extreme conditions if only regular operations are the concerning attributes during model building [113,168]. When attributes with less correlation are removed from the dataset, this process is called feature selection. Even though attributes with high correlation but irrelevant can cause overfitting in the model or negatively affect prediction performance by reducing the predictive accuracy [169, 170]. Because better models are generated or built when the input data are pre-processed correctly and efficiently, this shows how vital the pre-processing data stage of predictive model design can be in the real sense.

Another critical stage in the activity of predictive model design is data transformation. It consists of data discretization and normalization, among others. Data discretization involves using representative labels to replace numerical values, such as transforming periods into time bins. When values are bin helps them cope with the computational limitation of packages or optimize prediction models' training. In contrast, data normalisation reduces the values range to a well-defined range of values, such as 0 to 1 or from -1 to 1, to avoid inconsistencies, causing errors during model building. It helps equalise the variables' strengths and helps the Machine Learning methods be applied to identify which variables are more relevant. All these data pre-processing techniques are most important when using Machine Learning models for the predictive modelling task [83, 111, 171–174].

Grouping related items of datasets together such that the same group's data items are highly related but are dissimilar as possible from another group's data items is known as clustering. Research in the clustering area where different clustering methods, such as k-means, k-medoid and hierarchical, is used as a preliminary stage in the predictive model task [34, 37, 53, 66, 68, 147, 175–177].

2.3.4 Methods

There are several ways of modelling the passenger experience problem. It depends solely on the research objectives [35, 43, 50, 60, 143, 178–180]. Five significant categories of the method for solving passenger experience problems

concerning flight delay prediction, passenger flow prediction, or baggage/luggage handling deduced from the existing literature for this review are shown in Figure 2-5.

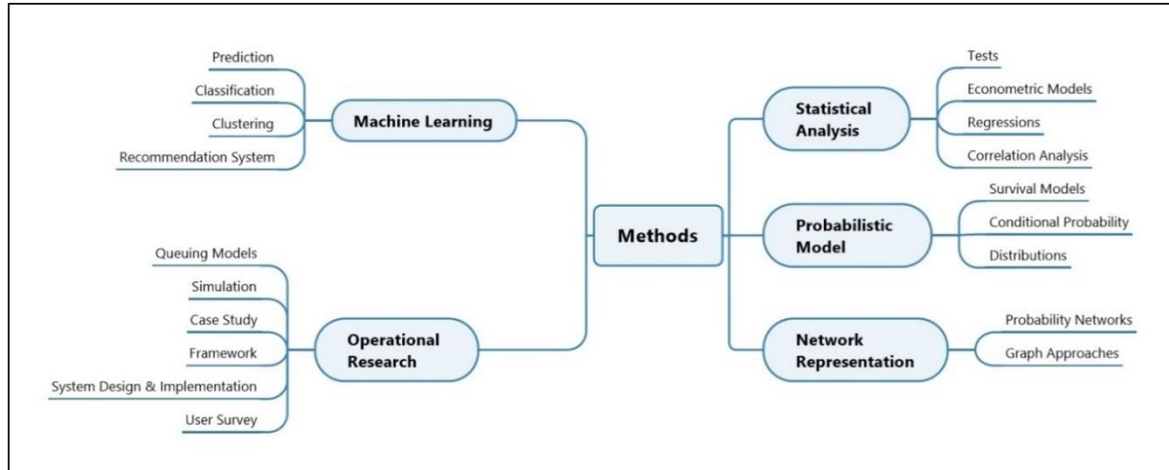


Figure 2-5 Categories of Methods for the Flight Delay Prediction Modelling

Operational Research

Operational research applies advanced analytics methods such as simulation, optimisation, and queue theory for better decision-making by the key players through experts in the field [181]. Airport capacity data may be analysed by simulation under different weather conditions for arrival and departure [50, 110, 187–190, 112, 127, 143, 182–186]. Each flight delay cost may be evaluated for an airline schedule improvement [56, 127]. Wieland applied simulation through queueing models [111, 147, 191]. Also, in root delay predictions by the study conducted by Kim and Hasan, Takeichi et al. [192, 125]. In contrast, others study airport demand and capacity effect on delay levels in New York by propagating delays between different airports [143, 174, 180].

Schedule stability and reliability of delay propagation were analyzed using simulation [132,187]. Under some external conditions, other researchers simulated the airport's reliability and flexibility in different scenarios. In contrast, other researchers solved the problems of congestion propagation and its effects in an airline on subsequent flights at Los Angeles International Airport by designing a simple deterministic queueing model that helped analyse and provide suggestions [14, 71, 81, 189, 193].

Network Representation

The study of flight systems through the application of graph theory is referred to as network representation. A model for airline schedule (resources available and time of flight) using a direct acyclic graph to study the impact of delays on other networks after detecting the delay's initial cause [194, 195]. The propagation's effect was evaluated using the classical shortest path algorithm. A propagation tree model was designed to compare two different airports by Ahmadbeygi et al., where one operates on a low-cost point-to-point system and the other on a conventional hub-and-spoke scheme system [140]. A Bayesian network for delay propagation was modelled by Xu et al. and Wu et al. [71, 182]. Historical data on flight-track of Europe was used to create a novel model for propagation delay using a network-epidemic [124, 197].

Probabilistic Model

Analysing event occurrence based on probability using analytical tools to estimate is considered a probabilistic model. An expectation-maximization with genetic algorithms to predict departure delay distribution in Denver International Airport based on a probabilistic model was designed, and congestion management [20, 145, 197–199]. A transition matrix was used to verify the delay propagation of subsequent flights and passengers' flow with a probabilistic mass function for expressing delay classes. The cancellation analysis can cancel a flight assuming a delay in the previous flight-based conditional probability [138, 200, 201]. Departure, arrival, *en route* delay, and service quality have been modelled by some authors using density functions [68, 111, 145]. Poisson distribution was verified best-fit arrival and *en route* delays, and Normal Distribution is best fitted for departure delays. A survival method was used to study the delay propagation to draw conclusions based on Wong and Tsai's total root delay propagation [169].

A model for integrating flight scheduling and routing was proposed using theoretical routing networks. In contrast, a model for the forecast restructuring of the US airline commercial network aims to reduce individual and total flight delays with algorithms [185, 202]. A model using the probability of whether or not there will be a blockage on the terminal route based on the convective weather forecast

and a simulation for airport runway capacity using Monte Carlo was built by some authors [203–205].

Statistical Analysis

So many models are under statistical analysis. They are as follows: Correlation analysis, regression model, non-parametric tests, parametric tests, econometric tests, univariate analysis (UVA) and multivariate analysis (MVA) [206]. Understanding the effect of delay propagation by the airlines through the network to help estimate the incurred cost can be done by both delay multipliers and recursive models using regression models [37, 52, 215, 216, 207–214]. The efficiency of flight and passenger flow monitoring systems is evaluated using econometric models to measure the government agency's investment level. Also, consider the existing relationship between passenger demand and delays, fares, and aircraft size and frequency by evaluating the equilibrium point of the items as mentioned above [220, 149]. An econometric model was designed to understand why it leads to flight cancellation among potential delay savings, pre-existing delays, and destination airlines' characteristics. The airport, aircraft size, frequency, fare and occupancy rate Xiong et al., while Qin et al. looked at the flight delay rates periodicity and Mofokeng et al. analysed the effect of turnaround during maintenance [71, 104, 218, 219]. Finally, a model that quantifies the originating delay point in New York and how it extends to other airports has been proposed by some authors.

In contrast, others try to check the impact of the capacity on the flow of passengers to avoid unnecessary costs [147, 220]. Other authors focused on statistical inferences. A non-parametric function was proposed to evaluate the efficiency of the United States airports regarding delays [221]. European airports were used to develop a model that computes the correlation between delay levels and capacities. The authors proposed different approaches for dealing with congestion, and the advantages and disadvantages of each were discussed [87, 112, 128, 190, 222]. Finally, the daily average of the principal causes of delays in the Orlando International Airport was proposed by calculating the daily delays and their correlations [223].

Machine Learning

The field of Machine Learning is built upon algorithms that can learn from data and provide predictions and insights for future instances. In recent years, there has been a surge of research focused on improving air passenger experience through the analysis of flight delays and passenger flow. Various Machine Learning methods, including k-means, neural networks, K-Nearest Neighbours, Support Vector Machines, ensemble methods, fuzzy logic, and Deep Learning, have been employed for tasks such as prediction, classification and segmentation [15, 108, 215, 231, 232, 224–230].

Ensemble methods, particularly random forest, have found wide application in tasks ranging from flight delay prediction to flight classification, passenger behavior prediction, and baggage handling. Comparative analysis of regression models over different time horizons within 24 hours has revealed an increase in prediction errors as the horizons extend. Researchers have also utilized adaptive network methods to examine the resource distribution nature among different airport components and monitor peak periods of delays within 24 hours. Additionally, predictive models for aircraft maintenance have been developed using these methods [104, 142, 198, 214, 237–242].

Reinforcement learning algorithms, such as Markov decision processes, have been employed for predicting taxi-out time delays. Impressive performance has been achieved in predicting delays at Tampa Bay International Airport and JFK International Airport in New York, with good results obtained 15 minutes before the occurrence of the delay .

Recommendation systems utilising the propagation delay effect have been constructed to forecast delays. K-Nearest Neighbour algorithms have been applied to identify similar events in historical data, demonstrating fast response times and simplicity as notable advantages of the model. Clustering techniques, particularly the k-means algorithm, have been employed to segment passengers based on airport data, allowing for a better understanding of diverse preferences among travelers [13, 18, 245].

Despite the widespread application of Machine Learning methods in passenger experience research, there remains a need for more novel

approaches that can reduce prediction errors and enhance the reliability of models. Future efforts should focus on developing more robust and accurate models to minimize additional risks associated with unreliable predictions [13, 18, 245–253, 26, 33, 67, 147, 176, 240, 243, 244].

2.4 Discussion of Results

There are several economic consequences caused to passengers, airlines and airports because of the problem related to passenger experience discussed in this review ranging from flight delay prediction, passenger flow prediction, or baggage/luggage handling failure, early awareness of the occurrence of any among them through predictions can go a long way to improve marketing decisions or even satisfaction of both customers and the airports/airlines service providers [49, 261–267]. Over the last decades, several models have been designed for prediction and forecast. This model aims to understand how flight delays, passenger flow problems, and baggage handling issues occur and their root causes and effects on the whole air transportation stakeholder, from the passengers down to the airports/airlines owner. Aside from these three points of looking at the passenger experience problem, the models for solving the problem could differ from the application scope, issues of data, and methods used to develop the models.

Figure 2-6 shows the relationship between the years and research on passenger experience because the number of articles increases as the year increases. In this particular analysis, consideration was not given to the specificity of application methods to build the solution from this review's identified list of methods. Therefore, given a particular year Y and the number of research articles N in passenger experience. We can formulate a relationship equation between them as written in Equation (2-1) and Equation (2-2), respectively:

$$Y \propto N \quad (2-1)$$

Simplifying further, we introduce a constant K to use equality in the relationship equation. The equation for the relationship can be written as follows.

$$Y = KN \quad (2-2)$$

Where Y is the year under consideration, and N is the number of research published in tackling passenger experience problems using the five major categories of methods identified in this review. K are external conditions that can be airport/airline passenger management, which means that passenger experience improvement has been a top priority of every air transport over the last decades. It will be worth noting that Equation (2-1) and Equation (2-2) are specifically applicable to the research conducted in this paper alone but not necessarily to other problems because the change in research directions in a particular research domain might affect the relationship negatively.

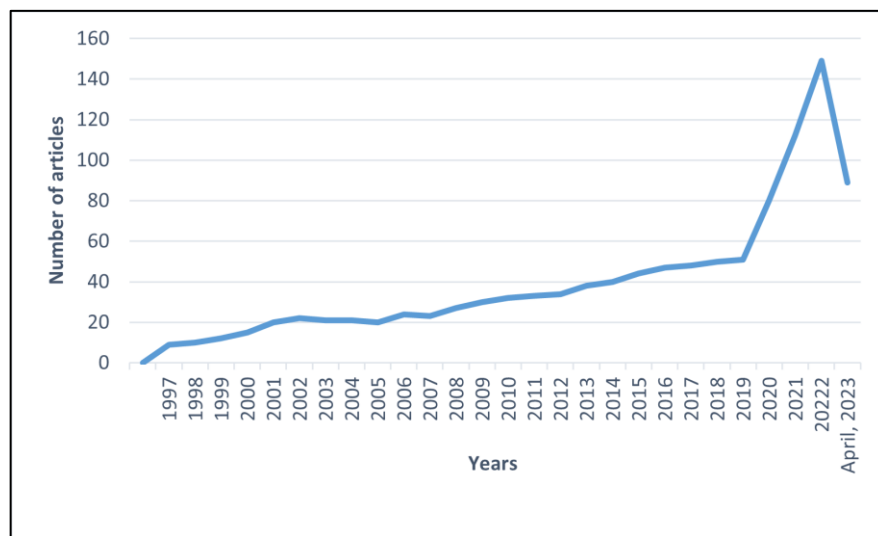


Figure 2-6 Relationship Between the Year and Passenger Experience Research Articles

Most of the papers were published in the late 2000s because 89.4% of the published articles were done between 2007 and 2022. In Figure 2-7a, there is the list of articles considered for the analysis in this review grouped according to methods where the massive growth in Machine Learning and data/data mining methods in the design of predictive models over the last decades can be observed [28, 105, 154, 268–274]. While Figure 2-7b shows how the methods have been applied in the three passenger experience problems considered for this review. Flight delay prediction has received more attention over the last decades. It is usually due to data available on the problems under consideration.

Most public airport and airline datasets contain flight and airport characteristics information. Simultaneously, passenger data are mostly kept

hidden because of the security attached to personal data except based on industry partners' requests. Research on passenger flow predictions is mainly generated using modelling or internet protocols at the airport to track passengers' movement within different time horizons using different scenarios and assumptions [2]. The research on baggage handling is only a few and the same reason as passenger flow predictions. From the literature studied, there has not been exact passenger or baggage data to the public like that of the flights and airlines records made available by different countries and regions, as earlier discussed in Table 2-2. On a balancing note, it is more apparent that flight delay has more work done while passenger flow and baggage handling has had little or no attention over the last decades [48, 95, 277, 98–100, 181, 185, 190, 275, 276].

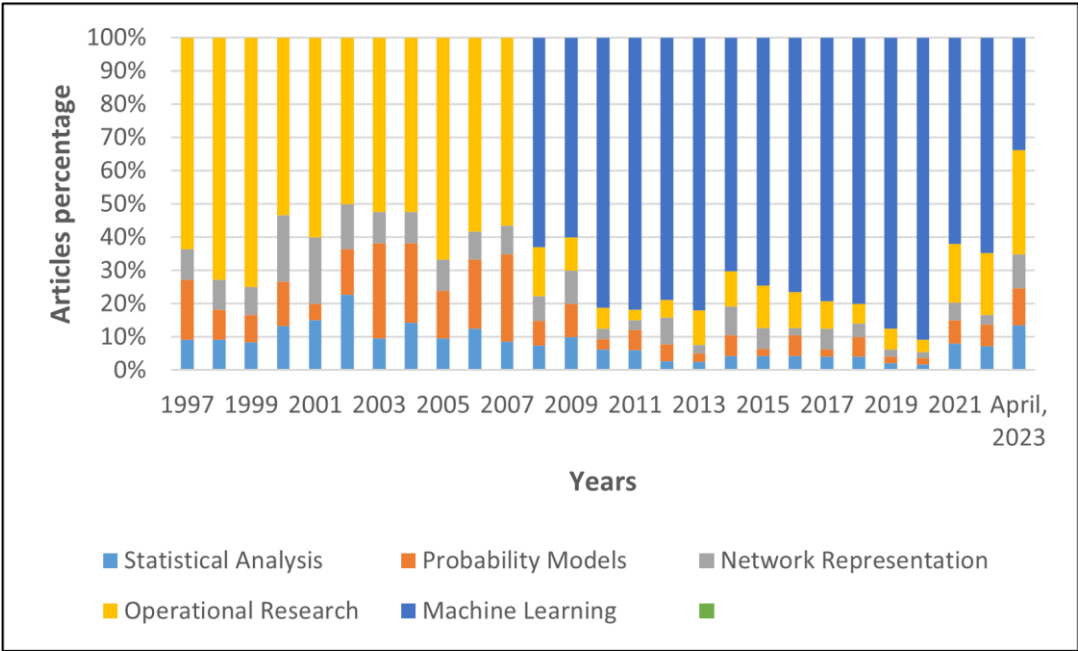


Figure 2-7a Publication on Main Methods According to Years

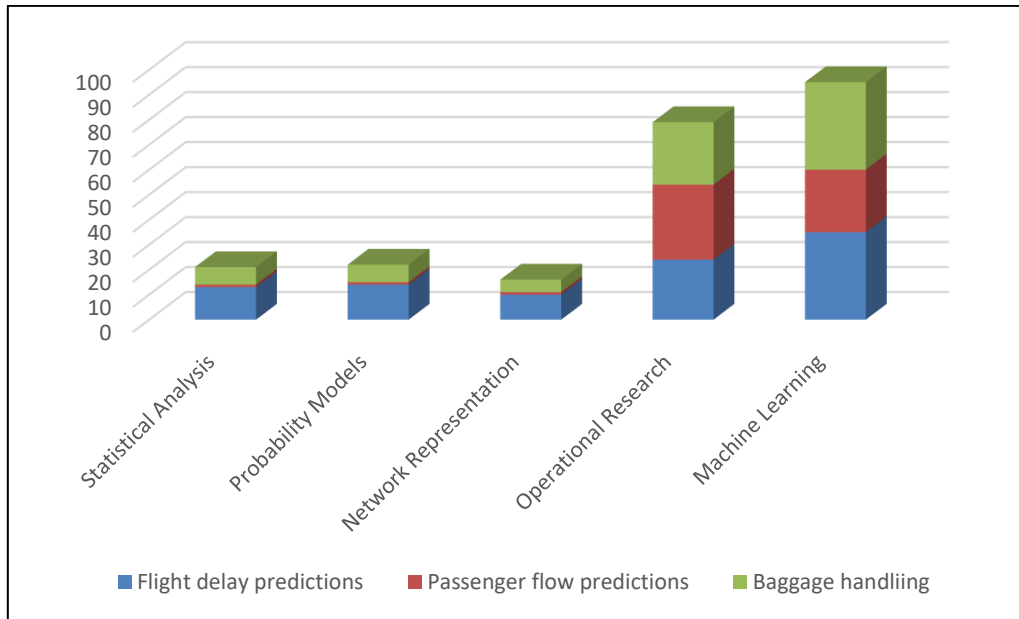


Figure 2-7b Methods Application to Problems

Figure 2-7a and Figure 2-7b show that operational research is the most applied method in solving passenger experience problems from the late 1990s, followed by probability models, then statistical, and finally, network representation. Not until the late 2000s has there not been a serious consideration of applying Machine Learning in solving passenger experience problems. Still, since it was introduced in our research, a constant increase has been noticed due to its high demand in industry solutions development because Machine Learning methods can train and learn from newly introduced instances during the model development [33, 37, 278–282, 84, 86, 89, 147, 174, 212, 252, 253].

Because our review is from a data science perspective of handling the passenger experience problem. The analysis is divided into three categories: the source of data, the data dimension, and the data management procedure. Based on the intense literature analysis in this review. We discover that when deciding which data source is being considered, most authors use data sources from the region or country where they are conducting the study. Some research based on data scope as a novelty used specific case study datasets from the concerned airports/airlines. For example, in the United States, most of the data source used in the research is from The United States Department of Transportation. China's

work is mostly on specific airport datasets [33, 35, 285, 286, 68, 69, 111, 147, 189, 252, 283, 284]. There was no direct relationship with the problem type or application scope of the problem from the data dimension. It is a notable characteristic in this case. Where capacity and demand, weathers were airport or *en-route* scope characteristics. Contrarily, airline elements were scope for airlines schedules. It is also worth noting that using different attribute combinations yields different results.

Management of data was not specific to any scope of application or problem. It has a steadily increasing use due to increased data generated from different aspects and applications that support passenger experience at the airport. Most Machine Learning models use data preparation techniques to transform and improve the data format. The probabilistic models and techniques consider outliers' removal for data transformations. Only a few percentages of the network representation, operational research, and statistical analysis methods used data management techniques. Historically, building predictive models, operational research, and statistical analysis were mostly used. They were spread across well between the three passenger experiences problems considered in this review. While probabilistic models have similar results, and network representation is mostly applied in the propagation of flight delays.

2.5 Conclusion, Summary of Findings and Recommendations

Airport/airline passenger experience problems from the perspective of (i) flight delay predictions, (ii) passenger flow predictions, and (iii) Baggage/Luggage handling is an essential subject of discussion from the literature because of its impact on the environment and economy of a country. It may increase operational costs, customer dissatisfaction, and integrity decline for airlines and airport operators. Aside from passenger-related issues, it is a crucial problem for the management during the decision-making process for all the players involved in the transportation system.

Based on this view, researchers have designed and proposed models over the last decade to handle any passenger mentioned above experience problems. This review contributed to conducting a study on the methods applied in solving passenger experience problems by grouping them into three unique groups in the

airport and airlines from data science and Machine Learning. A taxonomic analysis of methods applied over the last decades with detailed lists and explanations of specific methods under the general classes stated.

Two branches, such as data science and domain, were used to build the taxonomic analysis model for this review. Data science classifies the passenger experience problem into methods and data management. While the latter classifies the passenger experience into the domain and scope of the problem. Flight delay prediction was further classified into three major groups, which are (i) root delay, (ii) delay propagation and (iii) cancellation.

For the predictive model to improve accuracy requirements, we organized the problem's data aspect into sources, dimensions, and management supported by other researchers [25]. According to the studied literature, the methods were categorised into five classes for simplicity. This grouping was done based on how the predictive models were primarily designed for specific passenger experience problems.

From the taxonomy proposed, we design a view of the relationship between the publications and years of publications and the method applied to solve the problems. Future trends were also deduced that there would be more Machine Learning methods than other methods because the world is becoming more data-oriented. The high intelligence possessed by Machine Learning algorithms in learning from the vast generated data has helped them have a high level of accuracy and acceptance in general [287].

In conclusion, it was proven from our study that flight delay predictions had received more attention. It will be significant for airlines and airport managers to make policies that publish some data related to passengers, even if not in a traceable format. It will enable more work in passenger flow prediction and baggage handling to discover knowledge from the available Data. There is also a high need for researchers in academics and industries to build data science, Machine Learning and specialist skills. It is for the proper extraction of impactful hidden data knowledge through developing highly accurate models for handling future predictive problems of airports and airlines to improve passenger satisfaction.

2.5.1 Recommendations

Based on the study conducted in this review, the following recommendations and research directions are the possible research gaps that researchers in the future airport and airline passenger experience problems from the perspective of a flight delay, flow control, and handling baggage.

- i. The design of hybrid models for predictive tasks will be of good interest because a hybrid predictive model may introduce a new view and perspective in the Machine Learning and data research domain.
- ii. Research developing more sophisticated heterogeneous data integration methods can be considered since most study considers a single data view. Still, there is more increase in the amount of data generated daily than the world has ever experienced [288].
- iii. The airports and airlines should make available data about passengers and baggage on a non-identity view to work in crowd control and handle baggage no matter the level of failure in any of the processes. There is always massive damage to the passenger and the airports or airlines.

2.6 Research Gaps

This thesis identifies the following research gaps that need further advancements.

Research Gap 1: More intelligent algorithms are needed to extract the diverse hidden patterns and insights from the flight on-time performance datasets.

Research Gap 2: The need to extend the literature discussion of flight delays prediction.

Research Gap 3: More generalisable models for flight delay prediction are needed using Deep Learning, optimisation algorithms and meta-heuristic approaches.

Research Gap 4: The need to contextualise air passenger experience with reliable flight delay classification or prediction models.

2.7 References

1. Boksberger P. Aviation Systems. Aviation Systems. 2011; : 157–170. Available at: DOI:10.1007/978-3-642-20080-9
2. Papagiannopoulos N., Lopez JFG. Understanding and predicting passenger behaviours through data analytics. Journal of Airport

- Management. 2019; 13(1): 39–56.
3. de Juniac A. International Air Transport Association (IATA) - Annual review 2019. 75th Annual General Meeting, Seoul, June 2019. 2019; Available at: <https://www.iata.org/publications/Documents/iata-annual-review-2019.pdf> (Accessed: 3 May 2021)
4. Orsini F., Gastaldi M., Mantecchini L., Rossi R. Neural networks trained with WiFi traces to predict airport passenger behavior. MT-ITS 2019 - 6th International Conference on Models and Technologies for Intelligent Transportation Systems. IEEE; 2019; : 1–7. Available at: DOI:10.1109/MTITS.2019.8883365
5. Sternberg A., Soares J., Carvalho D., Ogasawara E. A review on flight delay prediction. 2017; : 1–21. Available at: <http://arxiv.org/abs/1703.06118>
6. Thiagarajan B., Srinivasan L., Sharma AV., Sreekanthan D., Vijayaraghavan V. A machine learning approach for prediction of on-time performance of flights. AIAA/IEEE Digital Avionics Systems Conference - Proceedings. 2017; 2017-Sept: 1–6. Available at: DOI:10.1109/DASC.2017.8102138
7. Pitchforth J., Wu P., Fookes C., Mengersen K. Processing passengers efficiently: An analysis of airport processing times for international passengers. Journal of Air Transport Management. Elsevier Ltd; 2015; 49: 35–45. Available at: DOI:10.1016/j.jairtraman.2015.06.016
8. Schultz M., Reitmann S. Machine learning approach to predict aircraft boarding. Transportation Research Part C: Emerging Technologies. 2019; 98: 391–408. Available at: DOI:10.1016/j.trc.2018.09.007
9. Schultz M. A metric for the real-time evaluation of the aircraft boarding progress. Transportation Research Part C: Emerging Technologies. Elsevier Ltd; January 2018; 86: 467–487. Available at: DOI:10.1016/j.trc.2017.11.002
10. Etani N. Development of a predictive model for on-time arrival flight of airliner by discovering correlation between flight and weather data. Journal of Data. Springer International Publishing; 2019; 6(1): 1–7. Available at: DOI:10.1186/s40537-019-0251-y
11. PeCoy MD., Redmond MA. Flight reliability during periods of high uncertainty. Journal of Air Transport Management. Elsevier Ltd; 2023; 106(August 2022). Available at: DOI:10.1016/j.jairtraman.2022.102322
12. Khanal A., Bhusal R., Subbarao K., Chakravarthy A., Okolo W. Learning Stochastic Processes Using Gaussian Processes: An Application to Flight Delay Prediction. 2023; (January): 1–13. Available at: DOI:10.2514/6.2023-1257
13. Chen S., Chen B., Shu P., Wang Z., Chen C. Real-time unmanned aerial vehicle flight path prediction using a bi-directional long short-term memory network with error compensation. Journal of Computational Design and Engineering. 2023; 10(1): 16–35. Available at: DOI:10.1093/jcde/qwac125
14. Jameel M. A Reference Point-Based Evolutionary Algorithm Solves Multi and Many-Objective Optimisation Problems : Method and Validation. 2023; 2023.
15. Khan R., Akbar S., Zahed TA. Flight Delay Prediction Based on Gradient Boosting Ensemble Techniques. IEEE; 2023; : 1–5. Available at:

- DOI:10.1109/icosst57195.2022.10016828
16. Tang Z., Huang S., Zhu X., Pan W., Han S., Gong T. Research on the multilayer structure of flight delay in China air traffic network. *Physica A: Statistical Mechanics and its Applications*. Elsevier B.V.; 2023; 609: 128309. Available at: DOI:10.1016/j.physa.2022.128309
 17. Liu C., Chen YR., Wang H., Zhang YY., Dai X., Luo Q., et al. Airport flight ground service time prediction with missing data using graph convolutional neural network imputation and bidirectional sliding mechanism. *Applied Soft Computing*. Elsevier B.V.; 2023; 133: 109941. Available at: DOI:10.1016/j.asoc.2022.109941
 18. Qu J., Wu S., Zhang J. Flight Delay Propagation Prediction Based on Deep Learning. *Mathematics*. 2023; 11(3): 494. Available at: DOI:10.3390/math11030494
 19. Piao C., Wang N., Yuan C. Rebalance Weights AdaBoost-SVM Model for Imbalanced Data. *Computational Intelligence and Neuroscience*. 2023; 2023: 1–26. Available at: DOI:10.1155/2023/4860536
 20. Faiza., Khalil K. Airline flight delays using artificial intelligence in COVID-19 with perspective analytics. *Journal of Intelligent & Fuzzy Systems*. 2023; : 1–23. Available at: DOI:10.3233/jifs-222827
 21. Ma X., He Z., Yang P., Liao X., Liu W. Agent-based modelling and simulation for life-cycle airport flight planning and scheduling. *Journal of Simulation*. Taylor & Francis; 2023; 00(00): 1–14. Available at: DOI:10.1080/17477778.2023.2169643
 22. Fatima SK., Campus A., Abbas SM., Campus A., Mir I., Campus A., et al. A Comprehensive Framework For Model Prediction of Aerial Vehicles : Perspective Analysis and Comparison. 2023; (January).
 23. Abdelghany A., Guzhva VS., Abdelghany K. The limitation of machine-learning based models in predicting airline flight block time. *Journal of Air Transport Management*. Elsevier Ltd; 2023; 107(November 2021): 102339. Available at: DOI:10.1016/j.jairtraman.2022.102339
 24. Civil Aviation Administration of China (CAAC). Statistics of Key Performance Indicators for China ' s Civil Aviation Industry in December 2022. 2022.
 25. Akpınar MT., Karabacak ME. Data mining applications in civil aviation sector: State-of-art review. *CEUR Workshop Proceedings*. 2017; 1852: 18–25. Available at: DOI:10.1016/j.apergo.2015.07.010
 26. Yu B., Guo Z., Asian S., Wang H., Chen G. Flight delay prediction for commercial air transport: A deep learning approach. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier; 2019; 125(March): 203–221. Available at: DOI:10.1016/j.tre.2019.03.013
 27. Ding Y. Predicting flight delay based on multiple linear regression. *IOP Conference Series: Earth and Environmental Science*. 2017; 81(1): 0–7. Available at: DOI:10.1088/1755-1315/81/1/012198
 28. Chakrabarty N. A data mining approach to flight arrival delay prediction for American airlines. *IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference*. 2019. pp. 102–107. Available at: DOI:10.1109/IEMECONX.2019.8876970
 29. Ye B., Liu B., Tian Y., Wan L. A methodology for predicting aggregate flight

- departure delays in airports based on supervised learning. *Sustainability*. 2020; 12(7): 2749. Available at: DOI:10.3390/su12072749
30. Lukas S., Stefani D., Widjaja P. Comparing SVM and GLM in calculating insurance premium for flight delay. *ACM International Conference Proceeding Series*. 2019. pp. 141–145. Available at: DOI:10.1145/3369114.3369160
 31. Baumgarten P., Malina R., Lange A. The impact of hubbing concentration on flight delays within airline networks: An empirical analysis of the US domestic market. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2014; 66: 103–114. Available at: DOI:10.1016/j.tre.2014.03.007
 32. D'Ariano A., Pistelli M., Pacciarelli D. Aircraft retiming and rerouting in vicinity of airports. *IET Intelligent Transport Systems*. 2012; 6(4): 433–443. Available at: DOI:10.1049/iet-its.2011.0182
 33. Bawa GK., Stansbury RS. A Study of Sector-based Weather Induced Flight Delay Using Deep Learning Techniques. 2022; (June). Available at: DOI:10.2514/6.2022-4080
 34. Balamurugan R., Maria A V., Baranidaran GT., Marygladence L., Revathy S. Error Calculation for Prediction of Flight Delays using Machine Learning Classifiers. 2022 6th International Conference on Trends in Electronics and Informatics, ICOEI 2022 - Proceedings. IEEE; 2022; (Icoei): 1219–1225. Available at: DOI:10.1109/ICOEI53556.2022.9776709
 35. Zhou F., Jiang G., Lu Z., Wang Q. Evaluation and Analysis of the Impact of Airport Delays. *Scientific Programming*. 2022; 2022(1). Available at: DOI:10.1155/2022/7102267
 36. Somani S., Priyanshu P., Meghna S., M S. An approach of applying machine learning model in flight delay prediction- A comparative analysis. *Revista Gestão Inovação e Tecnologias*. 2021; 11(3): 1233–1244. Available at: DOI:10.47059/revistageintec.v11i3.2006
 37. Lui GN., Thierry K., Liem RP. Data-driven approach for aircraft arrival flow investigation at terminal maneuvering area. *Aiaa Aviation 2020 Forum*. 2020; 1 PartF: 1–14. Available at: DOI:10.2514/6.2020-2869
 38. Bhadra D. You (expect to) get what you pay for: A system approach to delay, fare, and complaints. *Transportation Research Part A: Policy and Practice*. Elsevier Ltd; 2009; 43(9–10): 829–843. Available at: DOI:10.1016/j.tra.2009.08.003
 39. Dück V., Ionescu L., Kliwer N., Suhl L. Increasing stability of crew and aircraft schedules. *Transportation Research Part C: Emerging Technologies*. 2012; 20(1): 47–61. Available at: DOI:10.1016/j.trc.2011.02.009
 40. Fleurquin P., Campanelli B., Eguiluz VM., Ramasco JJ. Trees of reactionary delay: Addressing the dynamical robustness of the US air transportation network. *SIDs 2014 - Proceedings of the SESAR Innovation Days*. 2014.
 41. Valdes JJ., Cheung C., Li M. Sensor dynamics in high dimensional phase spaces via nonlinear transformations: Application to helicopter loads monitoring. *IEEE SSCI 2014 - 2014 IEEE Symposium Series on Computational Intelligence - CIDM 2014: 2014 IEEE Symposium on*

- Computational Intelligence and Data Mining, Proceedings. IEEE; 2015. pp. 365–372. Available at: DOI:10.1109/CIDM.2014.7008691
42. Ribeiro MF., Do U., Do V., Em M., Desenvolvimento AE., Ashenburg K., et al. Effect of total communication on academic achievement of Pupils with hearing impairment in Enugu State. Universidade Federal do Triângulo Mineiro. 2013. Available at: /citations?view_op=view_citation&continue=/scholar%3Fhl%3Dpt-BR%26as_sdt%3D0,5%26scilib%3D1&citilm=1&citation_for_view=wS0xi2wAAAAJ:2osOgNQ5qMEC&hl=pt-BR&oi=p
 43. Rodríguez-Sanz Á., Cano J., Rubio Fernández B. Impact of weather conditions on airport arrival delay and throughput. *Aircraft Engineering and Aerospace Technology*. 2022; 94(1): 60–78. Available at: DOI:10.1108/AEAT-12-2020-0318
 44. Fernandes N., Moro S., Costa CJ., Aparício M. Factors influencing charter flight departure delay. *Research in Transportation Business and Management*. Elsevier; 2019; 34(December): 100413. Available at: DOI:10.1016/j.rtbm.2019.100413
 45. Huang S-CA. Airline Delay Perturbation Problem Under Minor Disturbance. *National Center of Excellence for Nextor Aviation OPperation REearch*. 1998; 15(4): 250–260. Available at: DOI:10.11436/mssj.15.250
 46. Wohlin C. Guidelines for snowballing in systematic literature studies and a replication in software engineering. *ACM International Conference Proceeding Series*. 2014; Available at: DOI:10.1145/2601248.2601268
 47. Elisabeth A., Stacey L., Maintaining JD. Passenger experience in an airport : an activity-centred approach. *International Association of Societies of Design Research 2009 Proceedings: Design Rigor and Relevance*. Korean Society of Design Science, CD Rom, pp. 1-11. 2009. pp. 913–917.
 48. Liu X., Li L., Liu X., Zhang T., Rong X., Yang L., et al. Field investigation on characteristics of passenger flow in a Chinese hub airport terminal. *Building and Environment*. 2018; 133: 51–61. Available at: DOI:10.1016/j.buildenv.2018.02.009
 49. Sim KL., Koh HC., Shetty S. Some potential issues of service quality reporting for airlines. *Journal of Air Transport Management*. 2006; 12(6): 293–299. Available at: DOI:10.1016/j.jairtraman.2006.07.004
 50. Tan X., Jia R., Yan J., Wang K., Bian L. An Exploratory analysis of flight delay propagation in China. *Journal of Air Transport Management*. Elsevier Ltd; 2021; 92(February 2020): 102025. Available at: DOI:10.1016/j.jairtraman.2021.102025
 51. Xiong J., Hansen M. Modelling airline flight cancellation decisions. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2013; 56: 64–80. Available at: DOI:10.1016/j.tre.2013.05.003
 52. Horiguchi Y., Baba Y., Kashima H., Suzuki M., Kayahara H., Maeno J. Predicting Fuel Consumption and Flight Delays for Low-Cost Airlines. *Innovative Applications of Artificial Intelligence (IAAI) Conference*. 2017; : 4686–4693.
 53. Manna S., Biswas S., Kundu R., Rakshit S., Gupta P., Barman S. A statistical approach to predict flight delay using gradient boosted decision tree. *ICCIDS 2017 - International Conference on Computational*

- Intelligence in Data Science, Proceedings. 2018; 2018-Janua: 1–5. Available at: DOI:10.1109/ICCIDS.2017.8272656
54. Pamplona DA., Weigang L., De Barros AG., Shiguemori EH., Alves CJP. Supervised Neural Network with multilevel input layers for predicting of air traffic delays. Proceedings of the International Joint Conference on Neural Networks. IEEE; 2018; 2018-July. Available at: DOI:10.1109/IJCNN.2018.8489511
 55. Lambelho M., Mitici M., Pickup S., Marsden A. Assessing strategic flight schedules at an airport using machine learning-based flight delay and cancellation predictions. Journal of Air Transport Management. Elsevier Ltd; 2020; 82(September 2019): 101737. Available at: DOI:10.1016/j.jairtraman.2019.101737
 56. Zhu G., Matthews C., Wei P., Lorch M., Chakravarty S. En route flight time prediction under convective weather events. 2018 Aviation Technology, Integration, and Operations Conference. 2018; Available at: DOI:10.2514/6.2018-3670
 57. Huo J., Keung KL., Lee CKM., Ng KKH., Li KC. The prediction of flight delay: Data-driven machine learning approach. IEEE International Conference on Industrial Engineering and Engineering Management. 2020; 2020-Decem: 190–194. Available at: DOI:10.1109/IEEM45057.2020.9309919
 58. Guo Z., Yu B., Hao M., Wang W., Jiang Y., Zong F. A novel hybrid method for flight departure delay prediction using Random Forest Regression and Maximal Information Coefficient. Aerospace Science and Technology. Elsevier Masson SAS; 2021; 116: 106822. Available at: DOI:10.1016/j.ast.2021.106822
 59. Zhang W., Kamgarpour M., Sun D., Tomlin CJ. A hierarchical flight planning framework for air traffic management. Proceedings of the IEEE. IEEE; 2012; 100(1): 179–194. Available at: DOI:10.1109/JPROC.2011.2161243
 60. Livingston M., Zimmerman S. Prediction and Uncertainty Quantification of Daily Airport Flight Delays. 4th International Conference on Predictive Applications and APIs. 2018; August: 45–51.
 61. Nigam R., Govinda K. Cloud based flight delay prediction using logistic regression. Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2017. IEEE; 2018; (Iciss): 662–667. Available at: DOI:10.1109/ISS1.2017.8389254
 62. Dhanawade R., Deo M., Khanna N., Deolekar R V. Analyzing factors influencing flight delay prediction. Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development, INDIACom 2019. 2019; : 1003–1007.
 63. Kuhn N., Jamadagni N. Application of Machine Learning Algorithms to Predict Flight Arrival Delays. Cs229. 2017; : 1–6.
 64. Mayer C., Sinai T. Network effects, congestion externalities, and air traffic delays: Or why not all delays are evil. American Economic Review. 2003; 93(4): 1194–1215. Available at: DOI:10.1257/000282803769206269
 65. Cheng J. Estimation of flight delay using weighted Spline combined with ARIMA model. Proceedings of 2014 IEEE 7th International Conference on Advanced Infocomm Technology, IEEE/ICAIT 2014. IEEE; 2015; : 8–20.

- Available at: DOI:10.1109/ICAIT.2014.7019523
66. Chen Y., Yu J., Tsai SB., Zhu J. An Empirical Study on the indirect impact of flight delay on China's economy. *Sustainability* (Switzerland). 2018; 10(2). Available at: DOI:10.3390/su10020357
 67. McCarthy N., Karzand M., Lecue F. Amsterdam to Dublin Eventually Delayed? LSTM and Transfer Learning for Predicting Delays of Low Cost Airlines. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2019; 33: 9541–9546. Available at: DOI:10.1609/aaai.v33i01.33019541
 68. Gholami S., Khashe S. Alexa , Predict My Flight Delay. : 1–14.
 69. Wu W., Cai K., Yan Y., Li Y. An Improved SVM Model for Flight Delay Prediction. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. 2019; 2019-Septe. Available at: DOI:10.1109/DASC43569.2019.9081611
 70. Chen Z., Wanke P., Antunes JJM., Zhang N. Chinese airline efficiency under CO2 emissions and flight delays: A stochastic network DEA model. *Energy Economics*. Elsevier B.V.; 2017; 68: 89–108. Available at: DOI:10.1016/j.eneco.2017.09.015
 71. Gu Y., Yang J., Wang C., Xie G. Early warning model for passenger disturbance due to flight delays. *PLoS ONE*. 2020; 15(9 September): 1–13. Available at: DOI:10.1371/journal.pone.0239141
 72. Li Q., Guan C., Liu J. A CNN-LSTM Framework for Flight Delay Prediction. *Expert Systems with Applications*. Elsevier Ltd; 2023; : 120287. Available at: DOI:10.2139/ssrn.4283680
 73. Khanmohammadi S., Tutun S., Kucuk Y. A New Multilevel Input Layer Artificial Neural Network for Predicting Flight Delays at JFK Airport. *Procedia Computer Science*. The Author(s); 2016; 95(November): 237–244. Available at: DOI:10.1016/j.procs.2016.09.321
 74. Mokhtarimousavi S., Mehrabi A. Flight delay causality: Machine learning technique in conjunction with random parameter statistical analysis. *International Journal of Transportation Science and Technology*. Tongji Universit; 2023; 12(1): 230–244. Available at: DOI:10.1016/j.ijtst.2022.01.007
 75. Cao Y., Zhu C., Wang Y., Li Q. A Method of Reducing Flight Delay by Exploring Internal Mechanism of Flight Delays. *Journal of Advanced Transportation*. 2019; 2019(5): 22–25. Available at: DOI:10.1155/2019/7069380
 76. Cheevachaipimol W., Teinwan B., Chutima P. Flight delay prediction using a hybrid deep learning method. *Engineering Journal*. 2021; 25(8): 99–112. Available at: DOI:10.4186/ej.2021.25.8.99
 77. Muros Anguita JG., Díaz Olariaga O. Prediction of departure flight delays through the use of predictive tools based on machine learning/deep learning algorithms. *Aeronautical Journal*. 2023; 18(1): 1–23. Available at: DOI:10.1017/aer.2023.41
 78. Li Q., Jing R., Dong ZS. Flight Delay Prediction With Priority Information of Weather and Non-Weather Features. *IEEE Transactions on Intelligent Transportation Systems*. IEEE; 2023; 24(7): 7149–7165. Available at: DOI:10.1109/TITS.2023.3270743
 79. Yang Z., Chen Y., Hu J., Song Y., Mao Y. Departure delay prediction and analysis based on node sequence data of ground support services for

- transit flights. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2023; 153(August 2022): 104217. Available at: DOI:10.1016/j.trc.2023.104217
80. Sahadevan D., Al Ali H., Notman D., Mukandavire Z. Optimising Airport Ground Resource Allocation for Multiple Aircraft Using Machine Learning-Based Arrival Time Prediction. *Aerospace*. 2023; 10(6): 509. Available at: DOI:10.3390/aerospace10060509
 81. Sreenivasulu K., Sowjanya B., Motupalli VR., Yadav SH., Baseer KK., Pasha MJ. Prediction of Flight Delay through Intelligent Algorithms and Data Technology. 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC). IEEE; 2023; (Icaaic): 1074–1080. Available at: DOI:10.1109/icaaic56838.2023.10141246
 82. Birolini S., Jacquillat A. Day-ahead aircraft routing with data-driven primary delay predictions. *European Journal of Operational Research*. Elsevier B.V.; 2023; 310(1): 379–396. Available at: DOI:10.1016/j.ejor.2023.02.035
 83. Qu J., Chen B., Liu C., Wang J. Flight Delay Prediction Model Based on Lightweight Network ECA-MobileNetV3. *Electronics (Switzerland)*. 2023; 12(6). Available at: DOI:10.3390/electronics12061434
 84. Kockelman K., Chen TD., Street G. A Transportation Economics Reference for Practitioners: Final Report. 2013. Available at: <https://rosap.ntl.bts.gov/view/dot/25833>
 85. Karakostas B. Event prediction in an IoT environment using Naïve Bayesian models. *Procedia Computer Science*. Elsevier Masson SAS; 2016. pp. 11–17. Available at: DOI:10.1016/j.procs.2016.04.093
 86. Ferguson J., Kara AQ., Hoffman K., Sherry L. Estimating domestic US airline cost of delay based on European model. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2013; 33: 311–323. Available at: DOI:10.1016/j.trc.2011.10.003
 87. Moraffah R., Karami M., Guo R., Raglin A., Liu H. Causal Interpretability for Machine Learning - Problems, Methods and Evaluation. *ACM SIGKDD Explorations Newsletter*. 2020; 22(1): 18–33. Available at: DOI:10.1145/3400051.3400058
 88. Sreekumar K., Iyer V., Matalia K. Weather-based Aviation Delay Predictions. *International Journal of Recent Technology and Engineering*. 2019; (1): 2277–3878.
 89. Dukkanci O., Kara BY. Minimizing Energy and Cost in Range-Limited Drone Deliveries with Speed Optimisation. 2021; 125(January): 1–33. Available at: DOI:10.1016/j.trc.2021.102985
 90. Huang W., Lin Y., Lin B., Zhao L. Modelling and predicting the occupancy in a China hub airport terminal using Wi-Fi data. *Energy and Buildings*. Elsevier B.V.; 2019; 203. Available at: DOI:10.1016/j.enbuild.2019.109439
 91. Alom MZ., Taha TM., Yakopcic C., Westberg S., Sidike P., Nasrin MS., et al. The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches. 2018; Available at: <http://arxiv.org/abs/1803.01164>
 92. Al-Abri ES., S. E. Modelling atmospheric ozone concentration using machine learning algorithms. *PLoS ONE*. 2016; 13(3): e0194889. Available at: <https://dx.plos.org/10.1371/journal.pone.0194889>0Ahttps://dspace.lboro

- .ac.uk/dspace-jspui/handle/2134/25091
93. Learning M., Communications W. Machine Learning for Future Wireless Communications. Machine Learning for Future Wireless Communications. 2020. Available at: DOI:10.1002/9781119562306
 94. Lopez-Moreno I., Gonzalez-Dominguez J., Martinez D., Plchot O., Gonzalez-Rodriguez J., Moreno PJ. On the use of deep feedforward neural networks for automatic language identification. *Computer Speech and Language*. 2016; 40: 46–59. Available at: DOI:10.1016/j.csl.2016.03.001
 95. Ball M. O., Chen C. Y., Hoffman R. and VT. Collaborative decision making in air traffic management: Current and future research directions. In *New Concepts and Methods in Air Traffic Management*. 2001. Available at: DOI:10.1007/978-3-642-56680-6
 96. Yu R., Li Y., Shahabi C., Demiryurek U., Liu Y. Deep learning: A generic approach for extreme condition traffic forecasting. *Proceedings of the 17th SIAM International Conference on Data Mining, SDM 2017*. 2017; : 777–785. Available at: DOI:10.1137/1.9781611974973.87
 97. Lehouillier T., Soumis F., Omer J., Allignol C. Measuring the interactions between air traffic control and flow management using a simulation-based framework. *Computers and Industrial Engineering*. Elsevier Ltd; 2016; 99: 269–279. Available at: DOI:10.1016/j.cie.2016.07.025
 98. Chen X., He Z., Sun L. A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November 2018): 73–84. Available at: DOI:10.1016/j.trc.2018.11.003
 99. Burns MK., Ysseldyke JE. Reported prevalence of evidence-based instructional practices in special education. *Journal of Special Education*. 2009; 43(1): 3–11. Available at: DOI:10.1177/0022466908315563
 100. Jayam H., Nozick LK. Understanding the trade-off between maximum passenger throughput and airline equity in allocating capacity under severe weather conditions. *Transportation Research Record*. 2017; 2626: 18–24. Available at: DOI:10.3141/2626-03
 101. Sousa AMMLS de. Predictive Analysis of Flight Times. 2020.
 102. Natarajan V., Meenakshisundaram S., Balasubramanian G., Sinha S. A novel approach: Airline delay prediction using machine learning. *Proceedings - 2018 International Conference on Computational Science and Computational Intelligence, CSCI 2018*. IEEE; 2018. pp. 1081–1086. Available at: DOI:10.1109/CSCI46756.2018.00210
 103. Kim A., Hansen M. Deconstructing delay: A non-parametric approach to analyzing delay changes in single server queuing systems. *Transportation Research Part B: Methodological*. Elsevier Ltd; 2013; 58: 119–133. Available at: DOI:10.1016/j.trb.2013.10.001
 104. Hamzawi SG. Lack of airport capacity: Exploration of alternative solutions. *Transportation Research Part A*. 1992; 26(1): 47–58. Available at: DOI:10.1016/0965-8564(92)90044-8
 105. Weigang L., Alves CJP., Omar N. An expert system for air traffic flow management. *Journal of Advanced Transportation*. 1997; 31(3): 343–361. Available at: DOI:10.1002/atr.5670310308
 106. Bengio Y., Courville A., Vincent P. Representation learning: A review and

- new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. IEEE; 2013; 35(8): 1798–1828. Available at: DOI:10.1109/TPAMI.2013.50
107. Castaing J., Mukherjee I., Cohn A., Hurwitz L., Nguyen A., Müller JJ. Reducing airport gate blockage in passenger aviation: Models and analysis. *Computers and Operations Research*. Elsevier; 2016; 65: 189–199. Available at: DOI:10.1016/j.cor.2014.02.011
 108. Rodríguez-Sanz Á., Comendador FG., Valdés RA., Pérez-Castán J., Montes RB., Serrano SC. Assessment of airport arrival congestion and delay: Prediction and reliability. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November 2018): 255–283. Available at: DOI:10.1016/j.trc.2018.11.015
 109. Lawson D., Castillo W. Predicting flight delays. 2014; : 6. Available at: <http://cs229.stanford.edu/proj2012/CastilloLawson-PredictingFlightDelays.pdf>
 110. Liu L., Chen RC. A novel passenger flow prediction model using deep learning methods. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2017; 84: 74–91. Available at: DOI:10.1016/j.trc.2017.08.001
 111. Yap M., Luo D., Cats O., van Oort N., Hoogendoorn S. Where shall we sync? Clustering passenger flows to identify urban public transport hubs and their key synchronization priorities. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(December 2018): 433–448. Available at: DOI:10.1016/j.trc.2018.12.013
 112. Cheg L., Fookes C., Reddy V., Yarlagadda P. Analysis of passenger group behaviour and its impact on passenger flow using an agent-based model. 2014 4th International Conference On Simulation And Modelling Methodologies, Technologies And Applications (SIMULTECH). SCITEPRESS; 2014; : 733–738. Available at: DOI:10.5220/0005086807330738
 113. Liu, Xia, Fang Wan, Lei Chen, Zhao Qiu and MC. Research on traffic passenger volume prediction of Sanya city based on ARIMA and grey markov models. 4th International Conference of Pioneering Computer Scientists, Engineers and Educators Part II. Springer, Singapore; 2018. 69–74 p. Available at: DOI:10.30748/soi.2018.153.08
 114. Hunter G., Boisvert B., Ramamoorthy K. Advanced National Airspace traffic flow management simulation experiments and vlication. *Proceedings - Winter Simulation Conference*. 2007. pp. 1261–1267. Available at: DOI:10.1109/WSC.2007.4419730
 115. Baluch M., Bergstra T., El-Hajj M. Complex analysis of united states flight data using a data mining approach. 2017 IEEE 7th Annual Computing and Communication Workshop and Conference, CCWC 2017. IEEE; 2017. pp. 1–6. Available at: DOI:10.1109/CCWC.2017.7868414
 116. Qin G., Huang Z., Xiang Y., Sun J. ProbDetect: A choice probability-based taxi trip anomaly detection model considering traffic variability. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November 2018): 221–238. Available at: DOI:10.1016/j.trc.2018.11.016
 117. Ariyawansa CM., Aponso AC. Review on state of art data mining and

- machine learning techniques for intelligent Airport systems. Proceedings of 2016 International Conference on Information Management, ICIM 2016. IEEE; 2016. pp. 134–138. Available at: DOI:10.1109/INFOMAN.2016.7477547
118. Bunrit S., Kerdprasop N., Kerdprasop K. Improving the representation of cnn based features by autoencoder for a task of construction material image classification. *Journal of Advances in Information Technology*. 2020; 11(4): 192–199. Available at: DOI:10.12720/jait.11.4.192-199
 119. Mukhopadhyay S. *Advanced Data Analytics Using Python*. Advanced Data Analytics Using Python. Apress, Berkeley, CA; 2018. Available at: DOI:10.1007/978-1-4842-3450-1
 120. Wang Y. Prediction of weather impacted airport capacity using ensemble learning. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. IEEE; 2011; : 1–11. Available at: DOI:10.1109/DASC.2011.6096002
 121. Pai V. On the factors that affect airline flight frequency and aircraft size. *Journal of Air Transport Management*. Elsevier Ltd; 2010; 16(4): 169–177. Available at: DOI:10.1016/j.jairtraman.2009.08.001
 122. Evans AD., Sridhar B., McNally D. Improving operational acceptability of dynamic weather routes through analysis of commonly used routings. 16th *AIAA Aviation Technology, Integration, and Operations Conference*. 2016. pp. 1–18. Available at: DOI:10.2514/6.2016-3600
 123. Wan Y., Roy S. A scalable methodology for evaluating and designing coordinated air-traffic flow management strategies under uncertainty. *IEEE Transactions on Intelligent Transportation Systems*. 2008; 9(4): 644–654. Available at: DOI:10.1109/TITS.2008.2006813
 124. Schmidt M. A review of aircraft turnaround operations and simulations. *Progress in Aerospace Sciences*. 2017. Available at: DOI:10.1016/j.paerosci.2017.05.002
 125. Patel T., Chereesh C., Singh N., Somasekhar G. *Predicting Flight Delays using Machine Learning*. (V).
 126. United States Department of Transportation. Federal Aviation Administration. Washington: Office of Inspector General, U.S. Department of Transportation; 2020. Available at: <https://www.faa.gov/> (Accessed: 20 June 2021)
 127. United States Department of Transportation. Bureau of Transportation Statistics, Bureau of Transportation Statistics. United States Department of Transportation. 2020. Available at: <https://www.bts.gov/> (Accessed: 3 March 2020)
 128. Complete Streets | US Department of Transportation. 2020. Available at: <https://www.transportation.gov/mission/health/complete-streets>
 129. Airports | UK Civil Aviation Authority. Available at: <https://www.caa.co.uk/Environment/Environmental-information/Information-by-provider/Airports/>
 130. Datastore L. Flights and Passengers from London Airports - London Datastore. 2022. Available at: <https://data.london.gov.uk/dataset/flights-and-passengers-london-airports> (Accessed: 20 February 2022)
 131. Office for National Statistics. *International Passenger Survey: quality information in relation to migration flows* - Office for National Statistics.

- Office for National Statistics. 2017. Available at: <https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/internationalmigration/methodologies/internationalpassengersurveyqualityinformationinrelationtomigrationflows> (Accessed: 20 February 2022)
132. Kara K., Donna C., Katie L., Brice N. The Economics of Transportation Systems. The Future of Intermodal Freight Transport. 2013. Available at: DOI:10.4337/9781848441392
 133. Lee TTH., Yip TL. Impact of flight departure delay on airline choice behavior. *Asian Journal of Shipping and Logistics*. Elsevier B.V.; 2017; 33(4): 273–278. Available at: DOI:10.1016/j.ajsl.2017.12.011
 134. Kim AM. The impacts of changing flight demands and throughput performance on airport delays through the Great Recession. *Transportation Research Part A: Policy and Practice*. Elsevier Ltd; 2016; 86: 19–34. Available at: DOI:10.1016/j.tra.2016.02.001
 135. Yao S. Emerging Technologies in Data Mining and Information Security. In *Emerging Technologies in Data Mining and Information Security* (Vol. 813). <https://doi.org/10.1007/978-981-13-1498-8Secur>. Emerging Technologies in Data Mining and Information Security. Springer Singapore; 2019. 871–877 p. Available at: DOI:10.1007/978-981-13-1498-8
 136. Mou J., Liu C., Chen S., Huang G., Lu X. Temporal characteristics of the Chinese aviation network and their effects on the spread of infectious diseases. *Scientific Reports*. Springer US; 2017; 7(1): 1–9. Available at: DOI:10.1038/s41598-017-01380-5
 137. Takeichi N., Kaida R., Shimomura A., Yamauchi T. Prediction of delay due to air traffic control by machine learning. *AIAA Modelling and Simulation Technologies Conference*, 2017. 2017; (January): 1–8. Available at: DOI:10.2514/6.2017-1323
 138. Sternberg A., Carvalho D., Murta L., Soares J., Ogasawara E. An analysis of Brazilian flight delays based on frequent patterns. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2016; 95: 282–298. Available at: DOI:10.1016/j.tre.2016.09.013
 139. Soomer MJ., Franx GJ. Scheduling aircraft landings using airlines' preferences. *European Journal of Operational Research*. 2008; 190(1): 277–291. Available at: DOI:10.1016/j.ejor.2007.06.017
 140. Reynolds-Feighan AJ., Button KJ. An assessment of the capacity and congestion levels at European airports. *Journal of Air Transport Management*. 1999; 5(3): 113–134. Available at: DOI:10.1016/S0969-6997(99)00006-X
 141. Pejovic T., Williams VA., Noland RB., Toumi R. Factors affecting the frequency and severity of airport weather delays and the implications of climate change for future delays. *Transportation Research Record*. 2009; (2139): 97–106. Available at: DOI:10.3141/2139-12
 142. Pejovic T., Noland RB., Williams V., Toumi R. A tentative analysis of the impacts of an airport closure. *Journal of Air Transport Management*. Elsevier Ltd; 2009; 15(5): 241–248. Available at: DOI:10.1016/j.jairtraman.2009.02.004

143. Bubalo B., Daduna JR. Airport capacity and demand calculations by simulation-the case of Berlin-Brandenburg International Airport. *NETNOMICS: Economic Research and Electronic Networking*. 2011; 12(3): 161–181. Available at: DOI:10.1007/s11066-011-9065-6
144. Carr F., Theis G., Clarke JP., Feron E. Evaluation of improved pushback forecasts derived from airline ground operations data. *Journal of Aerospace Computing, Information and Communication*. 2005; 2(1): 25–43. Available at: DOI:10.2514/1.5730
145. Campanelli B., Fleurquin P., Eguíluz VM., Ramasco JJ., Arranz A., Etxebarria I., et al. Modelling reactionary delays in the European air transport network. *SIDs 2014 - Proceedings of the SESAR Innovation Days*. 2014.
146. Campanelli B., Fleurquin P., Arranz A., Etxebarria I., Ciruelos C., Eguíluz VM., et al. Comparing the modelling of delay propagation in the US and European air traffic networks. *Journal of Air Transport Management*. 2016; 56(Part A): 12–18. Available at: DOI:10.1016/j.jairtraman.2016.03.017
147. Sharath MN., Velaga NR., Quddus MA. A dynamic two-dimensional (D2D) weight-based map-matching algorithm. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(July 2018): 409–432. Available at: DOI:10.1016/j.trc.2018.12.009
148. Yazici MA., Kamga C., Singhal A. A data driven model for taxi drivers' airport pick-up decisions in New York City. *Proceedings - 2013 IEEE International Conference on Data, Data 2013*. IEEE; 2013. pp. 37–44. Available at: DOI:10.1109/BigData.2013.6691775
149. Vázquez IS. Air traffic flow management regulations: Data analytics. Universidad Carlos III de Madrid. Departamento de Bioingeniería e Ingeniería Aeroespacial; 2018. Available at: https://e-archivo.uc3m.es/bitstream/handle/10016/29552/TFG_Ignacio_Sanchez_Vazquez.pdf?sequence=1&isAllowed=y
150. Triantafillou E., Zhu T., Dumoulin V., Lamblin P., Evci U., Xu K., et al. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. *ICLR*. 2020. pp. 1–24. Available at: <http://arxiv.org/abs/1903.03096>
151. He L., Jiang D., Yang L., Pei E., Wu P., Sahli H. Multimodal affective dimension prediction using deep bidirectional long short-term memory recurrent neural networks. *AVEC 2015 - Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge, co-Located with MM 2015*. 2015; : 73–80. Available at: DOI:10.1145/2808196.2811641
152. AhmadBeygi S., Cohn A., Guan Y., Belobaba P. Analysis of the potential for delay propagation in passenger airline networks. *Journal of Air Transport Management*. 2008; 14(5): 221–236. Available at: DOI:10.1016/j.jairtraman.2008.04.010
153. Chiraphadhanakul V., Barnhart C. Robust flight schedules through slack re-allocation. *EURO Journal on Transportation and Logistics*. 2013; 2(4): 277–306. Available at: DOI:10.1007/s13676-013-0028-y
154. Balakrishna P., Ganesan R., Sherry L. Application of reinforcement learning algorithms for predicting taxi-out times. *Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009*. 2009; : 255–261.

155. Ciruelos C., Arranz A., Etxebarria I., Peces S., Campanelli B., Fleurquin P., et al. Modelling delay propagation trees for scheduled flights. Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2015. 2015. pp. 1–10. Available at: https://www.researchgate.net/profile/Victor-Eguiluz/publication/336837610_Modelling_Delay_Propagation_Trees_for_Scheduled_Flights/links/5db5bbe792851c577ecebb30/Modelling-Delay-Propagation-Trees-for-Scheduled-Flights.pdf
156. Fleurquin P., Ramasco JJ., Eguiluz VM. Systemic delay propagation in the US airport network. Scientific Reports. 2013; 3. Available at: DOI:10.1038/srep01159
157. Mueller ER., Chatterji GB. Analysis of aircraft arrival and departure delay characteristics. AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum. 2002. pp. 1–14. Available at: DOI:10.2514/6.2002-5866
158. Tu Y., Ball MO., Jank WS. Estimating flight departure delay distributions - A statistical approach with long-term trend and short-term pattern. Journal of the American Statistical Association. 2008; 103(481): 112–125. Available at: DOI:10.1198/016214507000000257
159. Zhang H., Song C., Wang H., Xu C., Guo J. Airport Delay Prediction Based on Spatiotemporal Analysis and Bi-LSTM Sequence Learning. Proceedings - 2019 Chinese Automation Congress, CAC 2019. 2019; (2017): 5080–5085. Available at: DOI:10.1109/CAC48633.2019.8996754
160. Penuel K., Statler M., Hagen R., Snow R. National Oceanic and Atmospheric Administration (NOAA). Encyclopedia of Crisis Management. 2013. Available at: DOI:10.4135/9781452275956.n222
161. Zou B., Hansen M. Flight delays, capacity investment and social welfare under air transport supply-demand equilibrium. Transportation Research Part A: Policy and Practice. Elsevier Ltd; 2012; 46(6): 965–980. Available at: DOI:10.1016/j.tra.2012.02.015
162. Zou B., Hansen M. Flight delay impact on airfare and flight frequency: A comprehensive assessment. Transportation Research Part E: Logistics and Transportation Review. Elsevier Ltd; 2014; 69: 54–74. Available at: DOI:10.1016/j.tre.2014.05.016
163. Zoutendijk M., Mitici M. Probabilistic flight delay predictions using machine learning and applications to the flight-to-gate assignment problem. Aerospace. 2021; 8(6): 152. Available at: DOI:10.3390/aerospace8060152
164. Raeis M., Tizghadam A., Leon-garcia A. Real-Time Prediction of Delay Distribution in Service Systems using Mixture Density Networks. : 1–14.
165. Sugara RA., Purwitasari D. Flight Delay Prediction for Mitigation of Airport Commercial Revenue Losses Using Machine Learning on Imbalanced Dataset. 2022; (November): 6–13.
166. López V., Fernández A., Herrera F. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. Information Sciences. Elsevier Inc.; 2014; 257: 1–13. Available at: DOI:10.1016/j.ins.2013.09.038
167. Witten H I., Frank E., Hall A M. Data Mining- Practical Machine Learning Techniques. Morgan Kaufmann. 2011. 584 p.

168. Carvalho L., Sternberg A., Maia Gonçalves L., Beatriz Cruz A., Soares JA., Brandão D., et al. On the relevance of data science for flight delay research: a systematic review. *Transport Reviews*. Taylor & Francis; 2021; 41(4): 499–528. Available at: DOI:10.1080/01441647.2020.1861123
169. Proenca HM., Klijn R., Bäck T., Van Leeuwen M. Identifying flight delay patterns using diverse subgroup discovery. *Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence, SSCI 2018*. 2019. pp. 60–67. Available at: DOI:10.1109/SSCI.2018.8628933
170. Bisandu DB., Homaid MS., Moulitsas I., Filippone S. A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective. *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK*. ACM; 2021. pp. 1–10. Available at: DOI:https://doi.org/10.1145/3505711.3505712
171. Makhloof MAA., Waheed ME., Badawi UAER. Real-time aircraft turnaround operations manager. *Production Planning and Control*. 2014; 25(1): 2–25. Available at: DOI:10.1080/09537287.2012.655800
172. Vandehzad M., Holmgren J. Efficient flight schedules with utilizing Machine Learning prediction algorithms. 2020;
173. Korfiatis N., Stamolampros P., Kourouthanassis P., Sagiadinos V. Measuring service quality from unstructured data: A topic modelling application on airline passengers' online reviews. *Expert Systems with Applications*. Elsevier Ltd; 2019; 116: 472–486. Available at: DOI:10.1016/j.eswa.2018.09.037
174. Şafak Ö., Atamtürk A., Selim Aktürk M. Accommodating new flights into an existing airline flight schedule. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 104(April): 265–286. Available at: DOI:10.1016/j.trc.2019.05.010
175. University of Westminster. The cost of passenger delay to airlines in Europe Consultation document University of Westminster. 2015; (August).
176. Zonglei L., Jiandong W., Guansheng Z. A new method to alarm large scale of flights delay based on machine learning. *Proceedings - 2008 International Symposium on Knowledge Acquisition and Modelling, KAM 2008*. IEEE; 2008. pp. 589–592. Available at: DOI:10.1109/KAM.2008.18
177. Kitchenham B., Brereton P. A systematic review of systematic review process research in software engineering. *Information and Software Technology*. Elsevier B.V.; 2013; 55(12): 2049–2075. Available at: DOI:10.1016/j.infsof.2013.07.010
178. Gupta D. Applied analytics through case studies using SAS and R: Implementing predictive models and machine learning techniques. *Applied Analytics through Case Studies Using SAS and R: Implementing Predictive Models and Machine Learning Techniques*. 2018. 1–404 p. Available at: DOI:10.1007/978-1-4842-3525-6
179. Rong Y., Wang J., Ding J. RIA-based visualization platform of flight delay intelligent prediction. *2009 Second ISECS International Colloquium on Computing, Communication, Control, and Management, CCCM 2009*. IEEE; 2009. pp. 94–97. Available at: DOI:10.1109/CCCM.2009.5267976
180. Le-Tien T., Nguyen-Thanh T., Xuan HP., Nguyen-Truong G., Ta-Quoc V.

- Deep learning based approach implemented to image super-resolution. *Journal of Advances in Information Technology*. 2020; 11(4): 209–216. Available at: DOI:10.12720/jait.11.4.209-216
181. Wong JT., Tsai SC. A survival model for flight delay propagation. *Journal of Air Transport Management*. Elsevier Ltd; 2012; 23: 5–11. Available at: DOI:10.1016/j.jairtraman.2012.01.016
 182. Wanke PF. Efficiency of Brazil's airports: Evidences from bootstrapped DEA and FDH estimates. *Journal of Air Transport Management*. Elsevier Ltd; 2012; 23: 47–53. Available at: DOI:10.1016/j.jairtraman.2012.01.014
 183. Evans JE. Analysis of Downstream Impacts of Air Traffic Delay. *Aviation*. 1997;
 184. Evans AD., Schäfer A., Dray L. Modelling airline network routing and scheduling under airport capacity constraints. 8th AIAA Aviation Technology, Integration and Operations (ATIO) Conference. 2008. pp. 1–13. Available at: DOI:10.2514/6.2008-8855
 185. Sridhar B., Chatterji G., Evans A. Lessons learned in the application of machine learning techniques to air traffic management. *Aiaa Aviation 2020 Forum*. 2020; 1 PartF: 1–12. Available at: DOI:10.2514/6.2020-2882
 186. Belkoura S., Peña JM., Zanin M. Beyond linear delay multipliers in air transport. *Journal of Advanced Transportation*. 2017; 2017. Available at: DOI:10.1155/2017/8139215
 187. Rebollo JJ., Balakrishnan H. Characterization and prediction of air traffic delays. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2014; 44: 231–241. Available at: DOI:10.1016/j.trc.2014.04.007
 188. Zhu X., Li L. Improved Flight Time Predictions for Fuel Loading Decisions of Scheduled Flights with a Deep Learning Approach. 2020; (11215119): 1–12. Available at: <https://arxiv.org/abs/2005.05684>
 189. Priyanka G. Prediction of Airline Delays Using K Nearest Neighbor Algorithm. 2018; 4(5): 87–90.
 190. Beatty R., Hsu R., Berry L., Rome J. Preliminary evaluation of flight delay propagation through an airline schedule. *Air Traffic Control Quarterly*. 1999; 7(4): 259–270. Available at: DOI:10.2514/atcq.7.4.259
 191. Ogunsina K., Okolo WA. Artificial Neural Network Modelling for Airline Disruption Management. 2021; Available at: DOI:10.2514/1.i011018
 192. Pyrgiotis N., Malone KM., Odoni A. Modelling delay propagation within an airport network. *Transportation Research Part C: Emerging Technologies*. 2013; 27: 60–75. Available at: DOI:10.1016/j.trc.2011.05.017
 193. Gatersleben MR. Analysis and simulation of passenger flows in an airport terminal. *Proceedings of the 31st conference on Winter simulation: Simulation---a bridge to the future-Volume 2*. 1999. 1999. pp. 1226–1231. Available at: DOI:10.1017/CBO9781107415324.004
 194. Alodhaibi S., Burdett RL., Yarlagadda PKDV. Framework for airport outbound passenger flow modelling. *Procedia Engineering*. 2017. pp. 1100–1109. Available at: DOI:10.1016/j.proeng.2017.01.263
 195. Xu N., Sherry L., Laskey KB. Multifactor model for predicting delays at U.S. airports. *Transportation Research Record*. 2008; (2052): 62–71. Available at: DOI:10.3141/2052-08
 196. Gu Y., Gowda GM., Jayanna PK., Boumghar R., Lucas L., Bernardi A., et

- al. The added value of advanced feature engineering and selection for machine learning models for spacecraft behavior prediction. 15th International Conference on Space Operations, 2018. 2018; (June): 1–11. Available at: DOI:10.2514/6.2018-2561
197. Evans A., Schäfer A. The rebound effect in the aviation sector. *Energy Economics*. Elsevier B.V.; 2013; 36: 158–165. Available at: DOI:10.1016/j.eneco.2012.12.005
 198. Hansen M., Zhang Y. Operational consequences of alternative airport demand management policies case of LaGuardia Airport, New York. *Transportation Research Record*. 2005; (1915): 95–104. Available at: DOI:10.3141/1915-12
 199. Wu CL. Inherent delays and operational reliability of airline schedules. *Journal of Air Transport Management*. 2005; 11(4): 273–282. Available at: DOI:10.1016/j.jairtraman.2005.01.005
 200. Zou B., Hansen M. Impact of operational performance on air carrier cost structure: Evidence from US airlines. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2012; 48(5): 1032–1048. Available at: DOI:10.1016/j.tre.2012.03.006
 201. Chung SH., Ma HL., Hansen M., Choi TM. Data science and analytics in aviation. *Transportation Research Part E: Logistics and Transportation Review*. 2020; 134(January). Available at: DOI:10.1016/j.tre.2020.101837
 202. Bogicevic V., Bujisic M., Bilgihan A., Yang W., Cobanoglu C. The impact of traveler-focused airport technology on traveler satisfaction. *Technological Forecasting and Social Change*. Elsevier; October 2017; 123(April): 351–361. Available at: DOI:10.1016/j.techfore.2017.03.038
 203. Kim D., Yang H. Evaluation of the risk frequency for hazards of runway incursion in Korea. *Journal of Air Transport Management*. Elsevier Ltd; 2012; 23: 31–35. Available at: DOI:10.1016/j.jairtraman.2012.01.011
 204. Farooq B., Bierlaire M., Hurtubia R., Flötteröd G. Simulation based population synthesis. *Transportation Research Part B: Methodological*. Elsevier Ltd; 2013; 58: 243–263. Available at: DOI:10.1016/j.trb.2013.09.012
 205. Abdelghany KF., Shah SS., Raina S., Abdelghany AF. A model for projecting flight delays during irregular operation conditions. *Journal of Air Transport Management*. 2004; 10(6): 385–394. Available at: DOI:10.1016/j.jairtraman.2004.06.008
 206. Kadhum M., Manaseer S., Dalhoum ALA. Evaluation feature selection technique on classification by using evolutionary ELM wrapper method with features priorities. *Journal of Advances in Information Technology*. 2021; 12(1): 21–28. Available at: DOI:10.12720/jait.12.1.21-28
 207. Baspinar B., Koyuncu E. A Data-Driven Air Transportation delay propagation model using epidemic process models. *International Journal of Aerospace Engineering*. 2016; 2016. Available at: DOI:10.1155/2016/4836260
 208. Wanke C., Song L., Zobell S., Greenbaum D., Mulgund S. Probabilistic congestion management. *Proceedings of the 6th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2005*. 2005. pp. 50–59.

209. Balakrishna P., Ganesan R., Sherry L. Accuracy of reinforcement learning algorithms for predicting aircraft taxi-out times: A case-study of Tampa Bay departures. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2010; 18(6): 950–962. Available at: DOI:10.1016/j.trc.2010.03.003
210. Muñoz A., Scarlatti D., Costas P. Real-time prediction of flight arrival times using surveillance information. *ACM International Conference Proceeding Series*. 2018; (Limc). Available at: DOI:10.1145/3241403.3241434
211. Pretty J., Benton TG., Bharucha ZP., Dicks L V., Flora CB., Godfray HCJ., et al. Global assessment of agricultural system redesign for sustainable intensification. *Nature Sustainability*. Springer US; 2018; 1(8): 441–446. Available at: DOI:10.1038/s41893-018-0114-0
212. Danermark B., Cieza A., Gangé JP., Gimigliano F., Granberg S., Hickson L., et al. International classification of functioning, disability, and health core sets for hearing loss: A discussion paper and invitation. *International Journal of Audiology*. 2010; 49(4): 256–262. Available at: DOI:10.3109/14992020903410110
213. Kotegawa T., De Laurentis D., Noonan K., Post J. Impact of commercial airline network evolution on the U.S. air transportation system. *Proceedings of the 9th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2011*. 2011. pp. 572–580.
214. Pfeil DM., Balakrishnan H. Identification of robust terminal-area routes in convective weather. *Transportation Science*. 2012; 46(1): 56–73. Available at: DOI:10.1287/trsc.1110.0372
215. Chen Z., Wang Y., Zhou L. Predicting weather-induced delays of high-speed rail and aviation in China. *Transport Policy*. Elsevier Ltd; 2021; 101(November 2020): 1–13. Available at: DOI:10.1016/j.tranpol.2020.11.008
216. Li Q., Jing R. Generation and prediction of flight delays in air transport. *IET Intelligent Transport Systems*. 2021; 15(6): 740–753. Available at: DOI:10.1049/itr2.12057
217. Zaporozhets OI. The quality of air transport services in function of improving the total quality of tourism offer. *Tour Hospitality Management*. 1997; 3(2): 381–392. Available at: DOI:10.1016/j.coviro.2013.07.007.Influenza
218. Rodríguez-Sanz Á., Fernández de Marcos A., Pérez-Castán JA., Comendador FG., Arnaldo Valdés R., París Loreiro Á. Queue behavioural patterns for passengers at airport terminals: A machine learning approach. *Journal of Air Transport Management*. 2021; 90(December 2019): 1–17. Available at: DOI:10.1016/j.jairtraman.2020.101940
219. Arif M., Gupta A., Williams A. Customer service in the aviation industry - An exploratory analysis of UAE airports. *Journal of Air Transport Management*. Elsevier Ltd; 2013; 32: 1–7. Available at: DOI:10.1016/j.jairtraman.2013.05.001
220. Cook Di., Lee EK., Majumder M. Data Visualization and Statistical Graphics in Data Analysis. *Annual Review of Statistics and Its Application*. 2016; 3: 133–159. Available at: DOI:10.1146/annurev-statistics-041715-033420
221. Maclejewski R., Pattath A., Ko S., Hafen R., Cleveland WS., Ebert DS.

- Automated box-cox transformations for improved visual encoding. *IEEE Transactions on Visualization and Computer Graphics*. IEEE; 2013; 19(1): 130–140. Available at: DOI:10.1109/TVCG.2012.64
222. Chandraa P., Prabakaran N., Kannadasan R. Airline delay predictions using supervised machine learning. *International Journal of Pure and Applied Mathematics*. 2018; 119(Special Issue 7A).
 223. Ravizza S., Chen J., Atkin JAD., Stewart P., Burke EK. Aircraft taxi time prediction: Comparisons and insights. *Applied Soft Computing Journal*. Elsevier B.V.; 2014; 14(PART C): 397–406. Available at: DOI:10.1016/j.asoc.2013.10.004
 224. Haralabopoulos G., Anagnostopoulos I., McAuley D. Ensemble deep learning for multilabel binary classification of user-generated content. *Algorithms*. 2020; 13(4): 1–14. Available at: DOI:10.3390/A13040083
 225. Tejasviram V., Solanki H., Ravi V., Kamaruddin S. Auto associative Extreme Learning Machine based non-linear principal component regression for data applications. *The 10th International Conference on Digital Information Management, ICDIM 2015*. IEEE; 2016. pp. 223–228. Available at: DOI:10.1109/ICDIM.2015.7381854
 226. Khan S., Yairi T. A review on the application of deep learning in system health management. *Mechanical Systems and Signal Processing*. Elsevier Ltd; 2018; 107: 241–265. Available at: DOI:10.1016/j.ymssp.2017.11.024
 227. Morrison SA., Winston C. The effect of FAA expenditures on air travel delays. *Journal of Urban Economics*. 2008; 63(2): 669–678. Available at: DOI:10.1016/j.jue.2007.04.005
 228. Xiong X., Zhang J., Sacker R., Ma J. Estimation and prediction on air flight delay status in the U.S and data visualization. 2017; (May). Available at: <https://search.proquest.com/docview/2155345563?pq-origsite=gscholar>
 229. Mofokeng TJ., Marnewick A. Factors contributing to delays regarding aircraft during A-check maintenance. 2017 IEEE Technology and Engineering Management Society Conference, TEMSCON 2017. 2017. pp. 185–190. Available at: DOI:10.1109/TEMSCON.2017.7998375
 230. Klein A., Kavoussi S., Lee RS. Weather forecast accuracy: Study of impact on airport capacity and estimation of avoidable costs. *Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009*. 2009. pp. 12–21.
 231. Pathomsiri S., Haghani A., Dresner M., Windle RJ. Impact of undesirable outputs on the productivity of US airports. *Transportation Research Part E: Logistics and Transportation Review*. 2008; 44(2): 235–259. Available at: DOI:10.1016/j.tre.2007.07.002
 232. Calderon R. Parental Involvement in Deaf Children's Education Programs as a Predictor of Child's Language, Early Reading, and Social-Emotional Development. *Journal of Deaf Studies and Deaf Education*. 2000; 5(2): 140–155. Available at: DOI:10.1093/deafed/5.2.140
 233. Abdel-Aty M., Lee C., Bai Y., Li X., Michalak M. Detecting periodic patterns of arrival delay. *Journal of Air Transport Management*. 2007; 13(6): 355–361. Available at: DOI:10.1016/j.jairtraman.2007.06.002
 234. Zhang X., Mahadevan S. Ensemble machine learning models for aviation incident risk prediction. *Decision Support Systems*. Elsevier; 2019;

- 116(October 2018): 48–63. Available at: DOI:10.1016/j.dss.2018.10.009
235. Hesam M., Sultana N., Millward H., Liu L. Ensemble learning activity scheduler for activity based travel demand models. *Transportation Research Part C. Elsevier Ltd*; 2021; 123(February): 102972. Available at: DOI:10.1016/j.trc.2021.102972
 236. Sataloff RT., Johns MM., Kost KM. Supervised and Unsupervised Ensemble Methods and their Applications.
 237. Sataloff RT., Johns MM., Kost KM. Ensembles in Machine Learning Applications.
 238. Galicia A., Talavera-Llames R., Troncoso A., Koprinska I., Martínez-Álvarez F. Multi-step forecasting for data time series based on ensemble learning. *Knowledge-Based Systems. Elsevier B.V.*; 2019; 163: 830–841. Available at: DOI:10.1016/j.knosys.2018.10.009
 239. Oliveira M., Torgo L. Ensembles for time series forecasting. *Journal of Machine Learning Research*. 2014; 39(2014): 360–370.
 240. Moretti F., Pizzuti S., Panzieri S., Annunziato M. Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modelling. *Neurocomputing. Elsevier*; 2015; 167: 3–7. Available at: DOI:10.1016/j.neucom.2014.08.100
 241. Dou X. Flight Arrival Delay Prediction And Analysis Using Ensemble Learning. 2020; (It nec): 836–840. Available at: DOI:10.1109/itnec48623.2020.9084929
 242. Korvesis P., Besseau S., Vazirgiannis M. Predictive maintenance in aviation: Failure prediction from post-flight reports. *Proceedings - IEEE 34th International Conference on Data Engineering, ICDE 2018. IEEE*; 2018; : 1423–1434. Available at: DOI:10.1109/ICDE.2018.00160
 243. Mode GR., Calyam P., Hoque KA. False Data Injection Attacks in Internet of Things and Deep Learning enabled Predictive Analytics. 2019; Available at: <http://arxiv.org/abs/1910.01716>
 244. Laine MHB., Puranik TG., Mavris DN., Matthews B. Predicting adverse events and their precursors in aviation using multi-class multiple-instance learning. *AIAA Scitech 2021 Forum*. 2021; 1 PartF(January): 1–21. Available at: DOI:10.2514/6.2021-0776
 245. Dangut MD., Skaf Z., Jennions IK. Rare Failure Prediction Using an Integrated Auto-encoder and Bidirectional Gated Recurrent Unit Network. *IFAC-PapersOnLine. Elsevier Ltd*; 2020; 53(3): 276–282. Available at: DOI:10.1016/j.ifacol.2020.11.045
 246. Rahaman MS., Hamilton M., Salim FD. Predicting imbalanced taxi and passenger queue contexts in airport. *Proceedings of the 21st Pacific Asia Conference on Information Systems: “Societal Transformation Through IS/IT”*, PACIS 2017. 2017. p. 172.
 247. George E., Khan SS. Reinforcement learning for taxi-out time prediction: An improved Q-learning approach. *2015 International Conference on Computing and Network Communications, CoCoNet 2015. IEEE*; 2016; : 757–764. Available at: DOI:10.1109/CoCoNet.2015.7411275
 248. Anwar A., Volkov M., Rus D. ChangiNOW: A mobile application for efficient taxi allocation at airports. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. IEEE*; 2013. pp. 694–701. Available at:

- DOI:10.1109/ITSC.2013.6728312
249. He Y., Hu M., Yuan L., Jiang H. Flight departure time prediction based on deep learning. 2022; : 23. Available at: DOI:10.1117/12.2623856
 250. Li B., Zhang D., Sun L., Chen C., Li S., Qi G., et al. Hunting or waiting? Discovering passenger-finding strategies from a large-scale real-world taxi dataset. 2011 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2011. IEEE; 2011; : 63–68. Available at: DOI:10.1109/PERCOMW.2011.5766967
 251. Wang X., Brownlee AEI., Woodward JR., Weiszer M., Mahfouf M., Chen J. Aircraft taxi time prediction: Feature importance and their implications. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2021; 124(November 2020): 102892. Available at: DOI:10.1016/j.trc.2020.102892
 252. Guo Q., Frosio I., Gallo O., Zickler T., Kautz J. Tackling 3D ToF Artifacts Through Learning and the FLAT Dataset. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2018; 11205 LNCS: 381–396. Available at: DOI:10.1007/978-3-030-01246-5_23
 253. Bisandu DB., Moulitsas I., Filippone S. Social ski driver conditional autoregressive-based deep learning classifier for flight delay prediction. *Neural Computing and Applications*. Springer London; 2022; 34(11): 8777–8802. Available at: DOI:10.1007/s00521-022-06898-y
 254. Zeng W., Li J., Quan Z., Lu X. A deep graph-embedded LSTM neural network approach for airport delay prediction. *Journal of Advanced Transportation*. 2021; 2021: 1–15. Available at: DOI:10.1155/2021/6638130
 255. Kim YJ., Choi S., Briceno S., Mavris D. A deep learning approach to flight delay prediction. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. IEEE; 2016. pp. 1–6. Available at: DOI:10.1109/DASC.2016.7778092
 256. Alharbi B., Prince M. A hybrid artificial intelligence approach to predict flight delay. *International Journal of Engineering Research and Technology*. 2020; 13(4): 814–822.
 257. Yazdi MF., Kamel SR., Chabok SJM., Kheirabadi M. Flight delay prediction based on deep learning and Levenberg-Marquart algorithm. *Journal of Data*. Springer International Publishing; 2020; 7(1). Available at: DOI:10.1186/s40537-020-00380-z
 258. Lin Y., Zhang J wei., Liu H. Deep learning based short-term air traffic flow prediction considering temporal–spatial correlation. *Aerospace Science and Technology*. Elsevier Masson SAS; 2019; 93: 105113. Available at: DOI:10.1016/j.ast.2019.04.021
 259. Liu H., Lin Y., Chen Z., Guo D., Zhang J., Jing H. Research on the Air Traffic Flow Prediction Using a Deep Learning Approach. *IEEE Access*. 2019; 7: 148019–148030. Available at: DOI:10.1109/ACCESS.2019.2945821
 260. Boggavarapu R., Agarwal P., Rohith Kumar DH. Aviation Delay Estimation using Deep Learning. 2019 4th International Conference on Information Systems and Computer Networks, ISCON 2019. IEEE; 2019; : 689–693.

- Available at: DOI:10.1109/ISCON47742.2019.9036276
261. Bezerra GCL., de Souza EM., Correia AR. Passenger expectations and airport service quality: Exploring customer segmentation. *Transportation Research Record*. 2021; 2675(10): 604–615. Available at: DOI:10.1177/03611981211011992
 262. Efthymiou M., Njoya ET., Lo PL., Papatheodorou A., Randall D. The impact of delays on customers' satisfaction: An empirical analysis of the british airways on-time performance at heathrow airport. *Journal of Aerospace Technology and Management*. 2019; 11: 1–13. Available at: DOI:10.5028/jatm.v11.977
 263. Kim S., Shin DH. Forecasting short-term air passenger demand using data from search engine queries. *Automation in Construction*. 2016; 70. Available at: DOI:10.1016/j.autcon.2016.06.009
 264. Daniel JI., Harback KT. (When) Do hub airlines internalize their self-imposed congestion delays? *Journal of Urban Economics*. 2008; 63(2): 583–612. Available at: DOI:10.1016/j.jue.2007.04.001
 265. Mukhopadhyay S., Mangope B., Moorad F. Voices of the voiceless: Inclusion of learners with special education needs in Botswana primary schools. *Exceptionality*. Routledge; 2019; 27(3): 232–246. Available at: DOI:10.1080/09362835.2018.1470446
 266. Ballakur AA., Arya A. Empirical evaluation of gated recurrent neural network architectures in aviation delay prediction. *Proceedings of the 2020 International Conference on Computing, Communication and Security, ICCCS 2020*. 2020; : 1–7. Available at: DOI:10.1109/ICCCS49678.2020.9276855
 267. Hutchinson D., Bogicevic V., Yang W., Bilgihan A., Bujisic M. Airport service quality drivers of passenger satisfaction. *Tourism Review*. 2013; 68(4): 3–18. Available at: DOI:10.1108/TR-09-2013-0047
 268. Shen S., Chen Z. Optimisation models for differentiating quality of service levels in probabilistic network capacity design problems. *Transportation Research Part B: Methodological*. Elsevier Ltd; 2013; 58: 71–91. Available at: DOI:10.1016/j.trb.2013.10.004
 269. Pantouvakis A., Renzi MF. Exploring different nationality perceptions of airport service quality. *Journal of Air Transport Management*. 2016; 52. Available at: DOI:10.1016/j.jairtraman.2015.12.005
 270. Keith, Cunningham; Kathryn, Cunningham and Joanne CO. Impact of differing cultural perceptions on special education service delivery. *Rural Special Education Quarterly*. 1988; 8(1): 2–8.
 271. Wang R., Shu-Li., Hsu., Lin YH., Tseng M-L. Evaluation of customer perceptions on airline service quality in uncertainty. *Procedia - Social and Behavioral Sciences*. Elsevier B.V.; 2011; 25(2011): 419–437. Available at: DOI:10.1016/j.sbspro.2012.02.054
 272. Benezech V., Coulombel N. The value of service reliability. *Transportation Research Part B: Methodological*. Elsevier Ltd; 2013; 58(2013): 1–15. Available at: DOI:10.1016/j.trb.2013.09.009
 273. Vonitsanos G., Panagiotakopoulos T., Kanavos A., Tsakalidis A. Forecasting Air Flight Delays and Enabling Smart Airport Services in Apache Spark. Springer International Publishing; 2021. 407–417 p.

- Available at: DOI:10.1007/978-3-030-79157-5
274. Lee K., Yu C. Assessment of airport service quality: A complementary approach to measure perceived service quality based on Google reviews. *Journal of Air Transport Management*. Elsevier Ltd; August 2018; 71(June): 28–44. Available at: DOI:10.1016/j.jairtraman.2018.05.004
 275. Gürbüz F., Özbakir L., Yapici H. Data mining and preprocessing application on component reports of an airline company in Turkey. *Expert Systems with Applications*. 2011; 38(6): 6618–6626. Available at: DOI:10.1016/j.eswa.2010.11.076
 276. Belcastro L., Marozzo F., Talia D., Trunfio P. Using scalable data mining for predicting flight delays. *ACM Transactions on Intelligent Systems and Technology*. 2016; 8(1): 1–20. Available at: DOI:10.1145/2888402
 277. Reddy Edla D., Lingras P., Venkatanareshbabu K., SpringerLink. *Advances in machine learning and data science: recent achievements and research directives*. Springer Singapore; 2018. 377 p.
 278. Kulkarni D., Wang Y., Sridhar B. Analysis of airport ground delay program decisions using data mining techniques. *AIAA AVIATION 2014 -14th AIAA Aviation Technology, Integration, and Operations Conference*. 2014. pp. 1–12. Available at: DOI:10.2514/6.2014-2025
 279. Ionescu L., Gwiggner C., Kliwer N. *Data Analysis of Delays in Airline Networks*. Business and Information Systems Engineering. Springer Fachmedien Wiesbaden; 2016; 58(2): 119–133. Available at: DOI:10.1007/s12599-015-0391-3
 280. Krawczyk B. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*. Springer Berlin Heidelberg; 2016; 5(4): 221–232. Available at: DOI:10.1007/s13748-016-0094-0
 281. Aljubairy A., Shemshadi A., Sheng QZ. Real-time investigation of flight delays based on the internet of things data. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2016; 10086 LNAI: 788–800. Available at: DOI:10.1007/978-3-319-49586-6_57
 282. Chen J., Li J. Airport passenger flow forecast based on the wavelet neural network model. In *Proceedings of the 2018 2Nd International Conference on Deep Learning Technologies*. 2018. pp. 61–65. Available at: DOI:10.1145/3234804.3234809
 283. Zaharia SE., Pietreanu CV. Challenges in airport digital transformation. *Transportation Research Procedia*. Elsevier B.V.; 2018; 35: 90–99. Available at: DOI:10.1016/j.trpro.2018.12.016
 284. Tikhonov AI., Sazonov AA., Novikov S V. Digital aviation industry in Russia. *Russian Engineering Research*. 2019; 39(4): 349–353. Available at: DOI:10.3103/S1068798X19040178
 285. Moreno-Torres JG., Saez JA., Herrera F. Study on the impact of partition-induced dataset shift on k-fold cross-validation. *IEEE Transactions on Neural Networks and Learning Systems*. 2012; 23(8): 1304–1312. Available at: DOI:10.1109/TNNLS.2012.2199516
 286. IATA. *AI in Aviation*. 2018; (June): 1–20. Available at: <http://www.iata.org/publications/Documents/AI-White-Paper.pdf>
 287. Spinielli E., Koelle R., Zanin M., Belkoura S. Initial implementation of

- reference trajectories for performance review. Proceedings of the 7th SESAR innovation days. Belgrade (Serbia). 2017. pp. 1–8.
288. Lillicrap TP., Hunt JJ., Pritzel A., Heess N., Erez T., Tassa Y., et al. Continuous control with deep reinforcement learning. 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings. 2016;
 289. Stefanovič P., Štrimaitis R., Kurasova O. Prediction of Flight Time Deviation for Lithuanian Airports Using Supervised Machine Learning Model. Computational Intelligence and Neuroscience. 2020; 2020. Available at: DOI:10.1155/2020/8878681
 290. Prentice C., Kadan M. The role of airport service quality in airport and destination choice. Journal of Retailing and Consumer Services. Elsevier Ltd; 2019; 47(July 2018): 40–48. Available at: DOI:10.1016/j.jretconser.2018.10.006
 291. Khanmohammadi S., Chou C., Iii HWL., Elias D. Airport. IEEE; 2014;
 292. Zheng YJ., Sheng WG., Sun XM., Chen SY. Airline Passenger Profiling Based on Fuzzy Deep Machine Learning. IEEE transactions on neural networks and learning systems. 2017; 28(12): 2911–2923. Available at: DOI:10.1109/TNNLS.2016.2609437
 293. Nguyen H., Kieu LM., Wen T., Cai C. Deep learning methods in transportation domain: A review. IET Intelligent Transport Systems. 2018; 12(9): 998–1004. Available at: DOI:10.1049/iet-its.2018.0064
 294. Hariri RH., Fredericks EM., Bowers KM. Uncertainty in data analytics: survey, opportunities, and challenges. Journal of Data. Springer International Publishing; 2019; 6(1). Available at: DOI:10.1186/s40537-019-0206-3

Chapter 3: Exploratory Data Analysis: A Flight On-Time Dataset View

Airport delays significantly influence passenger satisfaction with their overall travel experience. This research focuses on conducting an exploratory data analysis (EDA) of the flight on-time dataset from the US Bureau of Transportation Statistics (BTS), encompassing the top ten busiest airports in the United States, which serve millions of passengers annually. The dataset includes on-time performance during departure and flight arrival. Through EDA, we aim to uncover hidden knowledge and patterns within the data. The findings highlight the importance of EDA as the initial step in data analytics, enabling a deeper understanding of feature complexities and relationships before applying pre-processing techniques for Machine Learning algorithms. By leveraging EDA, valuable insights and hidden patterns within the dataset can be discovered, contributing to enhanced passenger experience and operational efficiency in the aviation industry.

3.1 Introduction

Visualisation of data and graphical statistics is an integration that fits between several disciplines. There is a startling in organising ideas and thoughts about what box-provoking uncertainty should reside. Graphical statistics primarily focus on visualising data with the understanding of variability. It can be crafted beautifully, elegantly, and interactive to tell hidden stories in data sets. Because of the importance of time in today's world, people prefer to utilise their time more judiciously during travelling by avoiding delays and too much waiting. Most passengers try to book a flight to avoid spending too much time at the airport.

Also, delays significantly impact the airport's entire time spent. The airport's performance is also affected by too much time spent at the airport, not only by the passengers. Furthermore, when a city has two airports, passengers will always prefer to use the less congested airports to avoid delays [1]. Losing a passenger means a fall in revenue for any airport. In airports and transport security and administration (TSA), one of the key performance indicators is the waiting time because 740 million passengers were expected to be screened in the USA in 2016 by the TSA [2,3]. Passenger satisfaction is not impacted by the wait time [4, 5]. There are other operations at the airport. With a higher waiting time, passengers are more within the airport facility, which means the facility running is increasing because of its challenges

[6–8]. There are unexpected delays that cascade delays that lead to high wait times and add pressure to more passengers on the staff's service level.

One of the most challenging aspects of Machine Learning model training and development is clearly understanding the data and how to define the problem [9–13]. This data clarity is essential to the process because it cannot be built successfully. Data exploration without any prior assumption reveals the ground truth about data content.

3.1.1 Deriving Intelligence from Data

Identifying the unique nature of the data under consideration is crucial. There are different data types, and each has an unconventional method of handling them before applying them to any model development. Different data stored by different disciplines contain different information and purposes. For example, aviation industries contain data strictly related to passengers' processes and activity, including sensor aircraft records. A dataset has different observations about the source from which it is generated. Our datasets contain numerical, discrete, continuous and categorical combinations.

- i. Numerical form: There is a sense of measurement in this data, for instance, departure/arrival time, actual elapsed time, and distance covered, among other essential features. It is also known as the quantitative data nature when used in statistics. It can be discrete or continuous. Discrete numerical data: This is listed and countable data. They are fixed and distinct. For example, the number of passengers on board could be 1,2,3,4, ..., n, while continuous numerical data is when data can have an infinite value within a specific range. Describe a variable that continues, for example, the delay between arrival and departure.
- ii. Categorical form: This represents object characteristics, for example, whether a flight is delayed or not. It is also known as qualitative data nature. There are several categorical data applications, such as male and female passengers. It contains limited values and is of different types, including binary or categorical polynomial data.

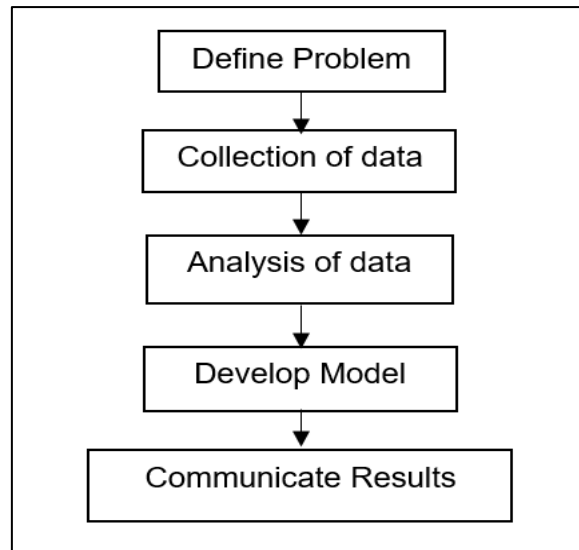


Figure 3-1 An Exploratory Data Framework

Airport passenger experience depends on all the aspects of the services provided in the airport. Still, due to limited methods and concentration on specific airports or airlines, the on-time flight delay records contain so much information currently underused in finding solutions to improve the airport passenger experience. This paper performs an exploratory analysis of heterogeneous flight on-time records to find hidden patterns that could be used to design Machine Learning predictive models. It is vital because of the best architecture to use in developing the model. After this stage of analysing the architecture development for testing, our data will be done [15,16].

Passengers will be encouraged to prefer air travel to other transportation means with sufficient knowledge of their travel plan [17, 18]. Hence, having a good exploration of data before data preparation will provide more insight into model training, provide more robust predictions for passengers during bookings, and allow passengers to spend time on other leisure activities. Also, reducing wait time makes an airport rate high sentiment during passengers' reviews. This paper analyses exploratory data on flight delay on-time records from the US BTS. This research presents a step to finding data patterns and presentation using Python programming with great data visualisation techniques.

3.1.2 The Problem

Data visualisation challenges have become more apparent with the increase in data from different activities by humans daily [17,18]. Good data plots from data are indispensably wrangled with the ability to be able to manipulate attributes accurately. Different philosophies about "data" exist depending on the person involved. However,

this paper looks at data problems from complexity rather than variable terms and the number of instances alone. It is because data of many variables are stored in an organised related table, perhaps a database with neglected massive records for years or data for business collected and stored in new systems such as Hadoop. Humans are potentially being informed about the world through data to improve business operations' learning efficiency, improve health care service delivery, improve tennis games to champion level through the correct statistics and Artificial Intelligence methods, and suggest the right network for social media.

The rest of the paper is organised as follows: Section 3.2 discusses related concepts, Section 3.3 describes datasets applied in this research, Section 3.4 presents the approach for analysis, Section 3.5 presents discussion and visualisation graphs with their comparison, section 3.6 discusses the limitation and future direction of this research and Section 3.7 contains the conclusion.

3.2 Related Work

The scarcity of papers on data visualisation cannot be overemphasised and is likely to be the most statistical journals candidates. Over time, data evolution has been highly interesting due to its importance in decision-making in all human living areas. They contain the candid picture of reality in their hidden formats until adequately mined. When dealing with time-oriented data, it is good to understand the inception for a better exploratory analysis. Insights and understanding of data are derived from detecting the patterns and trends in data. This analysis description does not override the fact that there is another essential visualisation task: checking the data element's existence, locating the elements in time, and checking the change rate.

Several data are time oriented. Stock exchange data is such data, simulation data, census data, and flight on-time data. However, temporal information can also be contained in photo collections, news articles, or project plans. Theoretically, the same approach can be used to represent all time-oriented datasets. Nevertheless, looking at it from a practical point of view, each data contains some uniqueness that makes them unique and will require a dedicated visualisation. For example, Flocking Boids is the best visualisation for stock exchange data [4,19–21]. Simulation data is better visualised with SimVis [4]. While ThemeRiver can be best for analysing news articles etc. When dealing with simple data and analysis, the mentioned methods outperform most specialised techniques due to their simplicity in learning and understanding.

However, time is considered a variable with one quantitate in many cases. For example, coordinates in parallel are not less except for their individual properties. Therefore, traditional methods do not establish direct connection visualisation of the time axis and multiple variables and allow direct exploration, interactions, and glancing of time data which is essential for a successful visualisation analysis.

Different approaches have extensively discussed visualising time series data in the literature. This paper applies the graphical statistics visualisation method for visualising time series [22–24]. The authors in [4] discussed an example of the calendar and cluster-based univariate time series data on different aggregations. The authors' survey discussed technical challenges with the constant increase in data and various visualisation forms based on information visualisation [17]. Using an exploratory analysis technique, the authors assessed a multinomial logit regression on passenger satisfaction levels and security screening for USA airports [2]. A model for airport security policies and operational planning performance, including wait time attributes and passenger satisfaction, was discussed [25–27]. [28, 29] discussed passenger arrival and departure flow based on general-purpose airport visualisation and simulation techniques. The authors in [32] extensively discussed a case study of managing passenger queues based on simulation models in an international airport in Australia. Many relevant examples of delays avoidance of days and days in a week based on the ten busiest airports in the USA were provided [30]. Most studies on exploratory data analysis motivate further exploration and analysis for more insights into the hidden knowledge contained in the data.

3.3 Datasets Description

The data used in this research to demonstrate our exploratory data analysis procedure is downloaded from the US Bureau of Transportation provided individual flight operations data for 2000, which was collected and extracted from October to December. For the month, the dataset collected included 1,048,575 flights from different airports. 52.08% of flights were on time, while 42.91% were delayed. Flights cancelled or diverted did not affect departed or delayed flights, so they were excluded from the dataset. Features such as Year, Destination, CRS Departure Time, Arrival Time, and Departure Delay are among the 29 attributes in the final dataset [31–33]. Table 3-1 shows the list of selected airports with their records of instances.

Table 1-1 Airports dataset information

S/No	Airports Codes	Airports Name
1	ATL	Hartsfield-Jackson Atlanta International Airport
2	ORD	O'Hare International Airport
3	LAX	Los Angeles International Airport
4	SAN	San Diego International Airport
5	SFO	San Francisco International Airport
6	PHL	Philadelphia International Airport
7	PHX	Phoenix Sky Harbor International Airport
8	DEN	Denver International Airport
9	MSP	Minneapolis-Saint Paul International Airport
10	BOS	Logan International Airport

The details of the instance's column are recorded and listed in Table 3-2. The records are clean of outliers and junks by filtering and replacing missing values through the integrated datasets record using Python programming language.

Table 3-2 Dataset information

S/No	Column	Units	Details
1	Year	Date	In four digits formats, e.g. 2000
2	Month	Date	In two digits formats, e.g. 12
3	Day of Month	Date	In two digits formats, e.g. 01-31
4	Day of Week	Date	In one-digit formats e.g. 1 (Monday) - 7 (Sunday)
5	Departure Time	Hours and Minutes	In four digits formats, e.g. 1456
6	Scheduled Departure Time	Hours and Minutes	In four digits formats, e.g. 1456
7	Arrival Time	Hours and Minutes	In four digits formats, e.g. 1456
8	Scheduled Arrival Time	Hours and Minutes	In four digits formats, e.g. 1456
9	Unique Carrier Code	Text	2 letters code, e.g. PS
10	Flight Number	Numeric	Number of flights, e.g. 1454
11	Plane Tail Number	AlphaNumeric	Plane tail number, e.g. N2343J
12	Actual Elapsed Time	Minutes	In two digits formats, e.g. 94
13	Scheduled Elapsed Time	Minutes	In two digits formats, e.g. 94
14	Flight Time	Minutes	In two digits formats, e.g. 94
15	Arrival Time	Minutes	In two digits formats, e.g. 94
16	Departure Time	Minutes	In two digits formats, e.g. 94
17	Origin	Text	3 letters code eg. LHR
18	Destination	Text	3 letters code, e.g. MAN
19	Distance	Numeric	The miles covered, e.g. 448
20	TaxiIn	Numeric	In two digits formats, e.g. 94

21	TaxiOut	Numeric	In two digits formats, e.g. 94
22	Cancelled	Binary	In two binary formats, e.g. 0 or 1
23	Cancellation Code	Text	In Alphabets formats, e.g. A, B, or C
24	Diverted	Binary	In two binary formats, e.g. 0 or 1
25	Carrier Delay	Minutes	In two digits formats, e.g. 94
26	Weather Delay	Binary	In two binary formats, e.g. 0 or 1
27	Aviation System Delay	Binary	In two binary formats, e.g. 0 or 1
28	Security Delay	Binary	In two binary formats, e.g. 0 or 1
29	Late Aircraft Delay	Binary	In two binary formats, e.g. 0 or 1

Selected Features and Importance

The dataset for input plays an essential role in the visualisation before performing the data preparation for Machine Learning to obtain an excellent performance from the model, and the data features selected can be broadly used to determine such performance. According to prior research, time-series features such as time and date are mainly included as selected visualisation features [52–55].

The basic time-related features and other important factors may affect the route flow. The entire flight delay data is a time series in nature. Therefore, the input vector x containing these features can be formed. Two types of features to define two vectors, which are impact vector s and time vector t , as in Equation (3-1) and Equation (3-2), can be written as follows:

$$\mathbf{t} = [t_1, t_2, t_3] \quad (3-1)$$

$$\mathbf{s} = [s_1, s_2, s_3] \quad (3-2)$$

t_1 , t_2 , and t_3 denote the hour of the day, the day of the week, and the day of the month. Furthermore, the holiday, season, and average flow index is denoted by s_1 , s_2 , and s_3 , respectively. Thus, the expression for input vector x , as shown in Equation (3-3), can be written as follows:

$$\mathbf{x} = [n, r, \mathbf{t}, \mathbf{s}] \quad (3-3)$$

The flight number is denoted by n within r the air route, and $\mathbf{t}$, $\mathbf{s}$, denotes the time and impact vectors.

The complicated relationship between the features and air route flow is non-linear and hard to extract using traditional methods, not dealing with the dataset. Thus, applying Artificial Intelligence techniques to massive datasets to better understand their relationships becomes necessary for a good Machine Learning discussion.

3.4 Approach for Analysis

Exploratory Data Analysis (EDA) is a critical procedure for preliminary data analyses to find patterns, identify anomalies, test hypotheses, and confirm presumptions [59]. EDA's main objective is to support data analysis before making any assumptions. It can help in the identification of obvious errors, the improvement of data patterns, the identification of outliers or uncommon events, and the identification of intriguing correlations between variables. The following inquiries regarding airports, airlines, and aircraft delays in the dataset were addressed using the EDA. Figure 3-3 shows a commercial airport's typical flight arrival and departure sequences. It indicates a typical step involved in the airport's processes even though some airports might have some specialise in additional steps on delivering their services. Information is processed faster and easier in the human brain using graphs or charts to visualise the complicated relationship and patterns of data [56, 57]. Data visualisation is a quick and easy concept conveyance in a generalised manner and different experiment scenarios by adjusting [58]. In this research, since dealing with data with time in different airports, glancing through the data will not provide patterns and observations to understand the entire operations' behaviour. Therefore, it is necessary to find the relation between the various data points using visualisation and graphs to help clean the data for Machine Learning models and pattern discovery. Several parameters impact delays, some of which are: extreme weather (0.72%), late-arriving aircraft (7.86%), air carrier (5.90%), National Aviation System (7.19%), and security (0.04%), as reported in 2017 by the Bureau of Transportation Statistics (BTS) on filing on-time performance of carriers in the USA [59–62], with a similar result in China [63,64].

To infer the patterns and trends from the enormous data available. We must analyse the data using the visualisation technique in Python. Section 5 contains the graphs and a detailed explanation of different attributes of the datasets and airports by visualisation approach, which captured the entire time series span to better explain the patterns hidden in the data [43]. It is chosen because of its robustness in dealing with time-series data to move pattern data in a vertical space fraction required by common area or line plots [65]. It is helpful to this research because it helps compare moving values of different types. Our work's benefit ranges from finding the possibilities of performing exploratory analysis on some of the most important

variables and factors contributing to the delay before applying a Machine Learning algorithm for model training. For instance, not all the variables were applied in the visualisation results. Still, the correlation shows the relationship level among all the variables. The following can be considered as the contributions made by this paper:

- i. We consider the possibility of analysing flight delay datasets and quantising some selected factors for the exploratory analyses. Therefore, we conduct visualisation analyses. The results show that data from different factors can be combined to perform exploratory analyses or develop flight delay prediction models.
- ii. Several visualisations of aviation datasets exist. Traditional exploratory analysis problem is mainly the statistical techniques on specific datasets. However, to comprehensively find the relationship and better generalisation of flight delay behaviour across the different factors, it is essential to consider integrating similar problems from different sources to perform the analysis for more insight [66–68].

3.5 Visualisation and Discussion

This section discusses the plots from the data to find the hidden insight into better interpretation and presentation towards Machine Learning modelling and airport business decision-making over a period showing the inter-relationships between different variables. Figure 3-2 shows the carriers' statistics summary, mostly occupying all the airlines with the green colour followed by the minimal delay having the orange colour and the extreme delay with the red colour. However, it is worth noting that the IATA condition for the delay is that any flight departing or arriving later than 15 Minutes (flights > 15 Minutes) is considered delayed. Therefore, Looking at the statistics combining both minimal and extreme delays of different carriers will almost reach the on time at some carriers. The UA has the dominant on-time records, followed by DL, AA, NW, EA, CO, PI, TW, WN, US, HP, PS, AS, and PA (1). The display is attributed to the number of flights that each carrier had when considering it. However, no relationship indicates when high on-time means the delays will be minimal; instead, otherwise. It indicates delays can grow if adequate measures are not set to control growth over time.

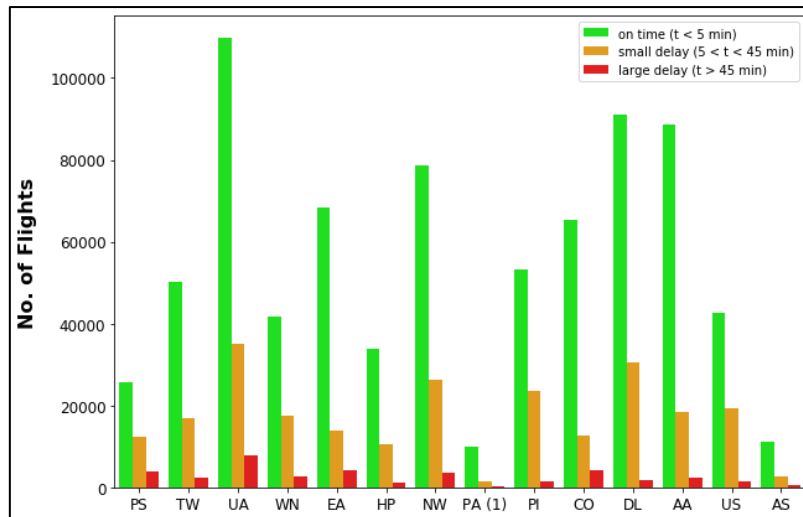


Figure 3-2 Statistics of Delays Among Unique Carriers

Figure 3-3 shows correlation matrix plots on the dataset's variables. It calculates the level of the relationships among all the existing variables in the dataset. It also gives a general overall insight into how related each variable is to another. It shows that in building a model, the highly correlated variables can monitor their performance across each other. That implies the impact of variables and building any model for either classification or predictive tasks. The two most crucial correlations of interest are the arrival and departure delays. We can see a high average correlation of 0.74 between arrival and departure delays. Despite more than 50% correlation, some flights arrive on time, even with the departure delay.

Nevertheless, it will be worth investigating the departure delay to see how the models' performance will subsequently affect the experiment. Also, the correlation between departure time and scheduled departure time is 0.92; the arrival time and the scheduled arrival time with a value of 0.87 mean an incredible impact. However, distance against actual elapsed time and scheduled elapsed time has the highest correlation of 0.95 and 0.98, meaning that the distance covered is affected by the elapsed time, respectively, on both actual and scheduled time.

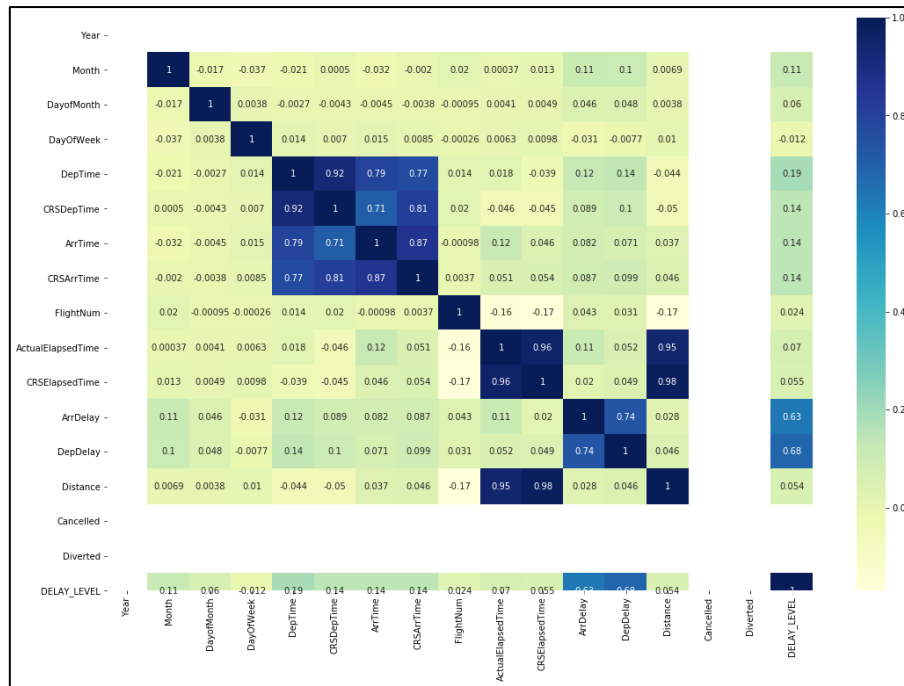


Figure 3-3 Correlation Matrix of Attributes

Figures 3-4 and 3-5 indicate the average delays of individual carriers (airlines) and arrival delay time distributions between different carriers. The plot's essence is to reveal the arrival delay incurred by the flights and the departure delays. The extent and patterns in which each carrier spread the delays across the period under consideration. In building the model, the delay from departure can be dropped to investigate by trying to predict and see how well the arrival can determine the departure delays. Classification can be run across our data to see how the models perform. It is worth seeing that most delays are around zero and five hundred minutes of the scheduled arrival time across all the carriers. The arrival delay's impact on the departure delay is high from the distribution of the plots of arrival delays on each unique carrier. It can contribute to up to 50 to 100 per cent of the entire departure delay in major cases when all other conditions are under control so as not to affect the operation of the airport's departure process.

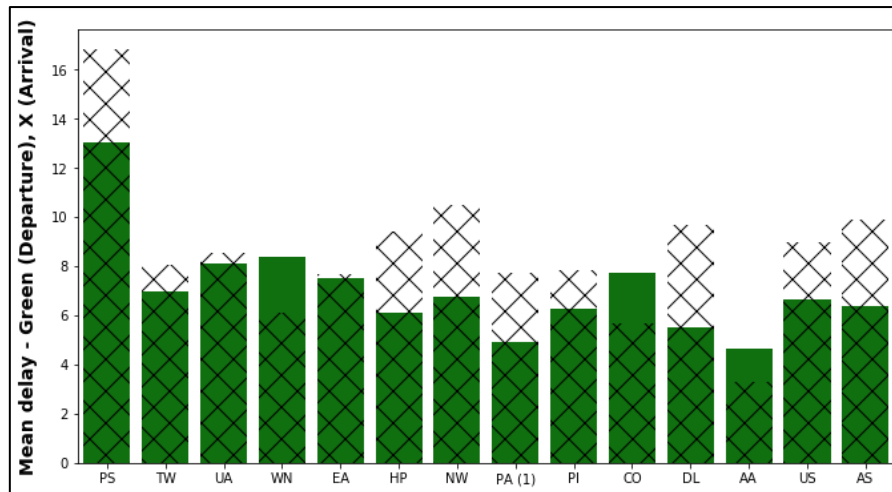


Figure 3-4 Mean Delays of Carriers

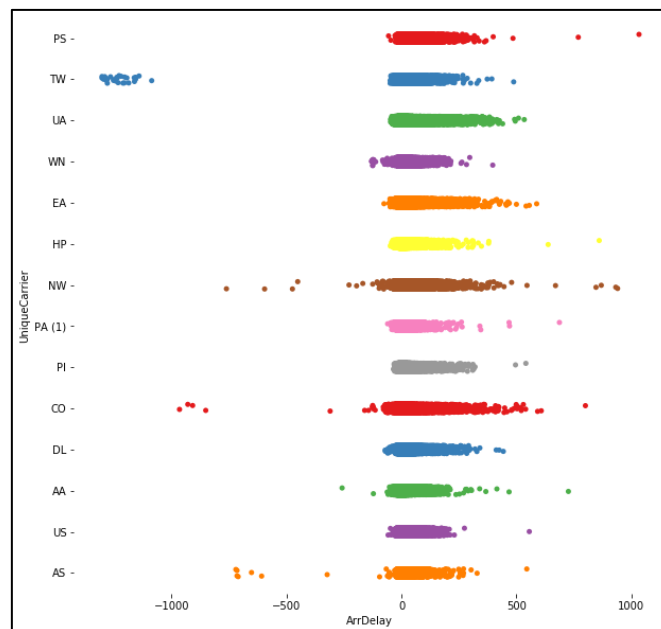


Figure 3-5 Arrival Delay Distribution of Carriers

Figure 3-6 shows the number of flights across different origin airports. It is significant because when looking at delays from an origin. The number of flights within the time of consideration in an airport can significantly affect either the departure or arrival time of flights due to failure in any process supporting the normal service flow during the travelling hours. The visualisation in Figure 3-8 will help manage resources and distribute them among the different airports because airports with more passengers will require more resources to manage them effectively and efficiently. The airports' managers are then provided with better insights from the data by just looking at it, which is one of the decisive advantages of visualisation statistics. They

can see how to increase or reduce resources across different services with less effort and time.

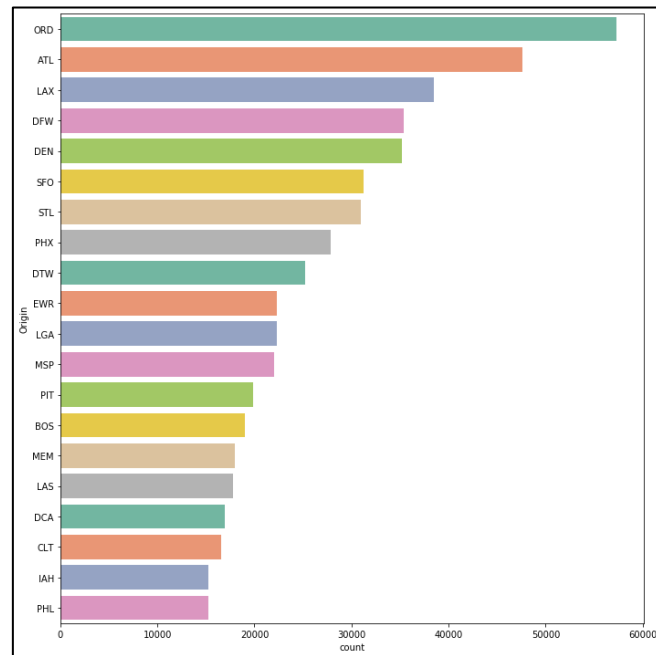


Figure 3.6 Flights Counts of Origin Airports

Figure 3-7 shows the box plot of delays and actual elapsed time within days of the week. The arrival delay occurs more within days 1 to 6 of the week, while the departure delay is spread from days 1 to 7, and the elapsed time is from days 3 to 7. This box distribution implies that it clarifies which week's period has high delays in arrival and departure to control human factors caused by the delay from airports and airlines. The arrival delay on departure delay is also evident because the departure delay exceeds the stage where the arrival delay stops.



Figure 3-7 Box Plot of Delays in a Week

United Airlines, Delta Airlines, American Airlines, Northwest Airlines, and Eastern Airlines are the top five airlines in terms of several flights operating in the United States, according to Figure 3-8.

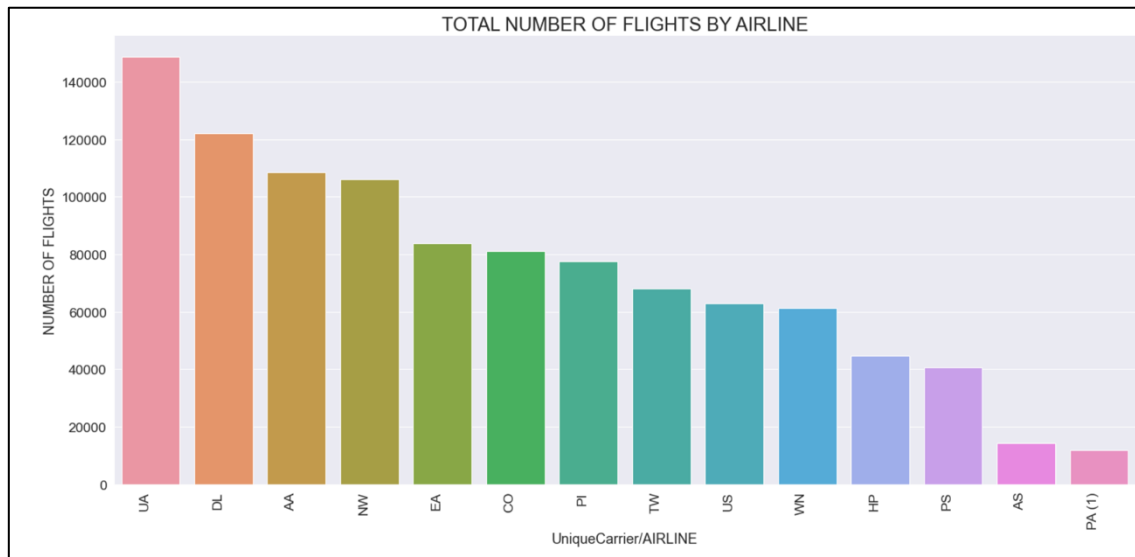


Figure 3-8 Total Number of Flight by Airline

Figure 3-9 shows that the airlines with the most delayed flights are Delta Airlines, United Airlines, Northwest Airlines and American Airlines.

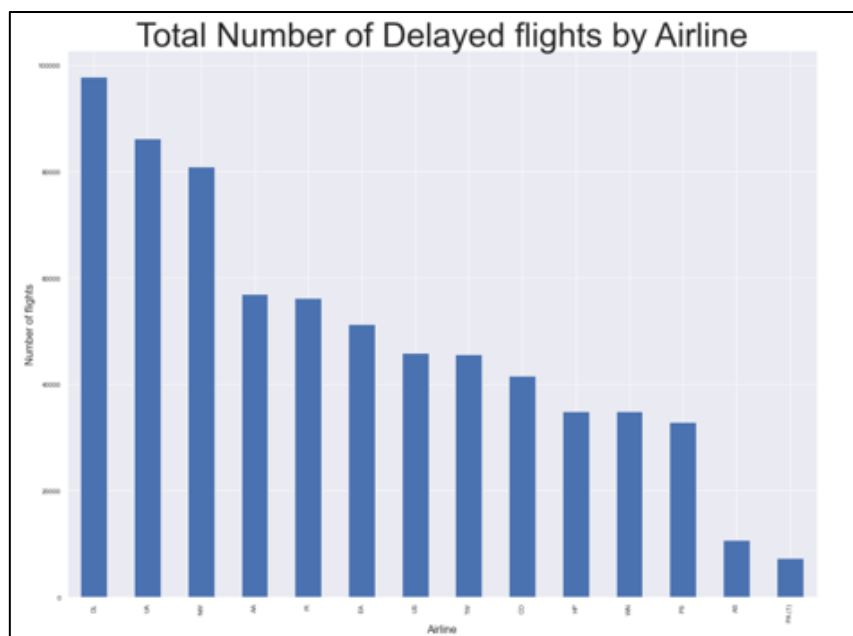


Figure 3-9 Total Number of Flights Delayed by Airlines

Of the 58 major airlines that fly domestic routes within the United States, only 13 are examined in the dataset [60, 61]. On the other hand, in Figure 3-10 red horizontal

line shows that in 2000, the average proportion of flights that were delayed over three months was 37.53%. Although all 13 carriers exceeded the threshold for delays, it's important to note that Delta and Northwest Airlines experienced the most delays despite being among the top 5 airlines in terms of flight volume. The fewest flights are delayed on American Airlines, United Airlines and Eastern Airlines.

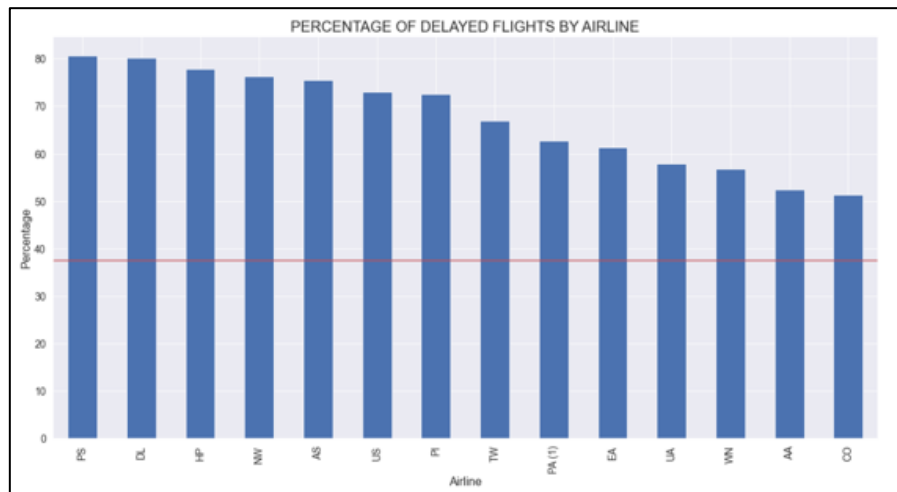


Figure 3-10 Airline Flight Delay Percentage

The most popular destinations have the largest arrival and departure delays out of 236 destination airports.

The top five destinations are Chicago (ORD), Atlanta (ATL), Los Angeles (LAX), Dallas-Fort Worth (DFW), and Denver (DEN). The average delay threshold on arrival for the top 30 destinations is 5 minutes, as seen by the horizontal dotted line across the graph in Figure 3-11. However, the graphs reveal that LAX and DEN have a high average delay, indicating that if you fly to both airports, there is a good possibility your flight will be delayed. On the other hand, DFW is the 4th most popular destination, with a very short arrival wait time. The city with the most delays out of the top 30 destinations is by far San Francisco (SFO), with an average of 20mins, where you are practically certain to arrive late. The same can be said for other cities.

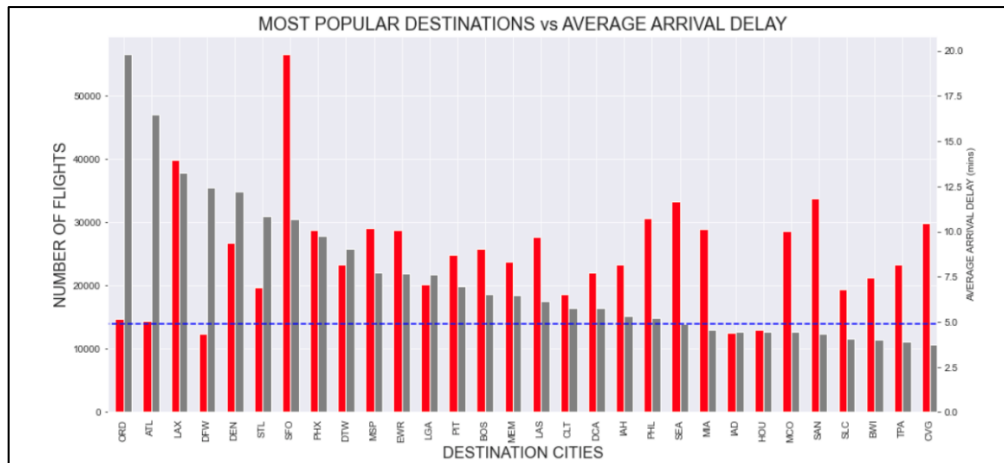


Figure 3-11 Most Popular Destinations - Arrival Delay

Figure 3-12 shows a graph that compares the most popular destinations with average departure delays, with the blue dotted line representing the average departure time in minutes. The average delay in departure time is between 10 to 11 minutes, with ORD being the most popular destination, with the longest delays surpassing 14 minutes. In contrast, we can infer that flights departing or arriving in Chicago try to compensate for late departures by shortening the elapsed time, which appears to work on average. Conversely, ATL is still in a good position as the second most popular destination, with a low arrival delay and an average departure delay of fewer than 10 minutes. SFO is once again distinct, with the city with the highest average of departure delays.

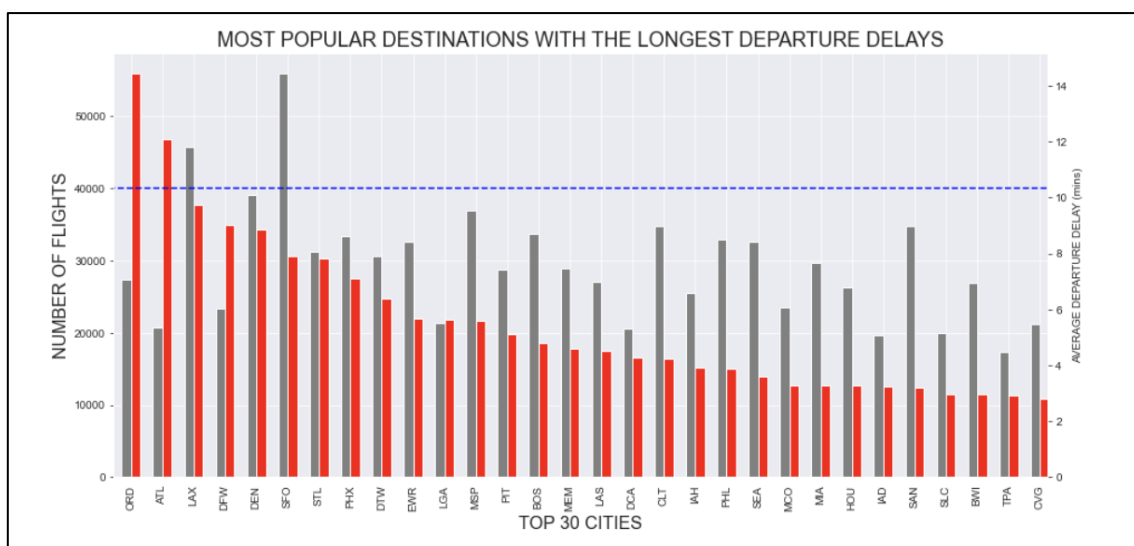


Figure 3-12 Most Popular Cities with Departure Delays

Figure 3-13 shows the number of destinations per airline, with United Airlines being the biggest of the 13 domestic airlines in our dataset. Here are a few overviews of everything we know about UA so far. We know that it is the largest airline in the number of flights. It has one of the fewest flight delays and one of the lowest percentages of delayed flights amongst domestic and commercial airlines. It indicates they perform well and would be America's most reliable and recommended airline.

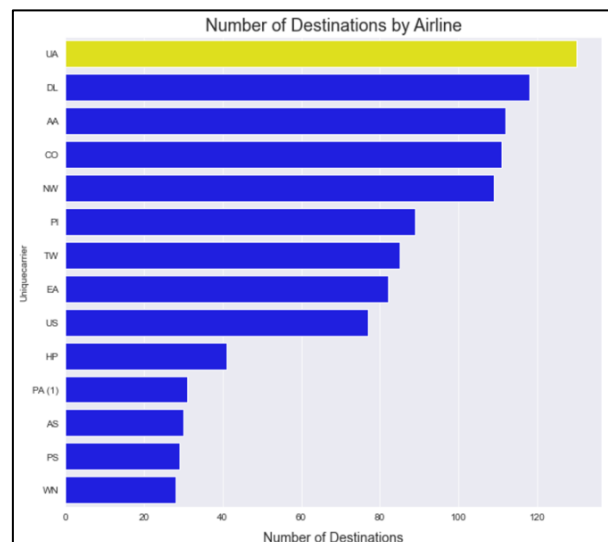


Figure 3-13 Number of Destinations by Airline

Figure 3-14 shows the top 20 airports and the flight delays in minutes. Again, San Francisco tops the chart with an average delay of 19 minutes; this could also be due to the large flights in the airport.

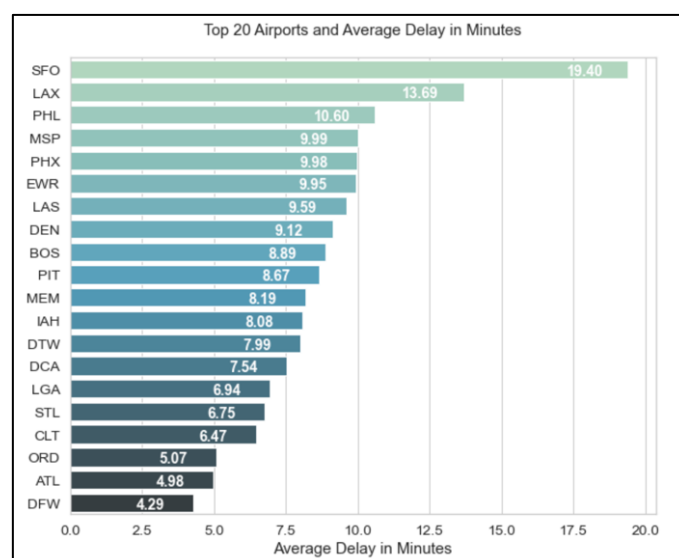


Figure 3-14 Top Airports and Average Delay

3.6 Limitations and Future Directions

This research's limitation is that data combination might negatively impact the visualisation of results and the statistical correlation between the datasets whole variables. Also, the data variables have not been considered to visualise their entire distribution because we only try to answer the question related to the investigation we proposed in the delay problem at the arrival and departure. Visualisation cannot reveal the depth of the hidden patterns in a dataset. It can only be a first step to analysing datasets. Therefore, the predictive and prescriptive model analysis will be needed to derive more hidden insights from the data. Dataset for capacity planning and operational constraints such as airports waiting for layout queues priority and space utilisation for booths were not considered, which might have also influenced the visualisation results.

The future extension of the paper can be in any of the following areas:

- i. Predictive modelling: Regression models for arrival or departure delays prediction can be built from the current exploration analysis given the details like arrival time, scheduled arrival time, elapsed time, and actual elapsed times, and such models will help in developing hybrid predictive systems for flight delays which can help passengers in having full knowledge of their trips for better planning.
- ii. Prescriptive modelling: Aside from “predictive modelling” with regression, using some non-linear optimisation models to suggest flight timing using the current behaviour of all the parameters needed. Performing prescriptions through prescriptive modelling will help airports maintain lower delays and reduce customer dissatisfaction through flight delays.

3.7 Conclusion

An approach to visualising flight delays from different data parameters influencing them is also identified for performing a prescriptive or predictive approach through the correlation matrix statistics. One of the key outcomes expected from reducing flight delays is passenger satisfaction. Therefore, visualising delay patterns is essential for thoroughly analysing the consequences of their occurrence. Next is investigating each airport from its delay monitoring systems. All the various delays can be connected as a unified system towards a universal, robust delay predictive system for the predictions to be independent of a particular region being less biased. However, each airport

needs to understand the peculiarity of delays clearly. A high generalisation model for all the airports is possible, leading to more accurate predictions if carefully integrated. Prescribing the number of delays across different airports using models would solve the problem of optimising the non-linear complexity of the delay prediction system. It is not straightforward, and each airport needs to be carefully studied before building the hybrid model. The extension of this research can be to visualise the arrival and departure of more airports integrated to reveal the delay patterns.

3.8 References

1. Sternberg A., Soares J., Carvalho D., Ogasawara E. A review on flight delay prediction. 2017;: 1–21. Available at: <http://arxiv.org/abs/1703.06118>
2. Sankaranarayanan HB., Agarwal G., Rathod V. An exploratory data analysis of airport wait times using data visualisation techniques. 2016 International Conference on Computation System and Information Technology for Sustainable Solutions, CSITSS 2016. 2016. pp. 324–329. Available at: DOI:10.1109/CSITSS.2016.7779379
3. Cook Di., Lee EK., Majumder M. Data Visualization and Statistical Graphics in Data Analysis. Annual Review of Statistics and Its Application. 2016; 3: 133–159. Available at: DOI:10.1146/annurev-statistics-041715-033420
4. Aigner W., Miksch S., Müller W., Schumann H., Tominski C. Visualizing time-oriented data-A systematic view. Computers and Graphics (Pergamon). 2007; 31(3): 401–409. Available at: DOI:10.1016/j.cag.2007.01.030
5. Rong Y., Wang J., Ding J. RIA-based visualisation platform of flight delay intelligent prediction. 2009 Second ISECS International Colloquium on Computing, Communication, Control, and Management, CCCM 2009. IEEE; 2009. pp. 94–97. Available at: DOI:10.1109/CCCM.2009.5267976
6. Gkiotsalitis K., Wu Z., Cats O. A cost-minimisation model for bus fleet allocation featuring the tactical generation of short-turning and interlining options. Transportation Research Part C: Emerging Technologies. 2019; 98(November 2018): 14–36. Available at: DOI:10.1016/j.trc.2018.11.007
7. Li B., Zhang D., Sun L., Chen C., Li S., Qi G., et al. Hunting or waiting? Discovering passenger-finding strategies from a large-scale real-world taxi dataset. 2011 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2011. IEEE; 2011; : 63–68. Available at: DOI:10.1109/PERCOMW.2011.5766967
8. Rossi R., Gastaldi M., Orsini F. How to drive passenger airport experience: A decision support system based on user profile. IET Intelligent Transport Systems. 2018; 12(4): 301–308. Available at: DOI:10.1049/iet-its.2017.0210
9. Tejasviram V., Solanki H., Ravi V., Kamaruddin S. Auto associative Extreme Learning Machine based non-linear principal component regression for data applications. The 10th International Conference on Digital Information Management, ICDIM 2015. IEEE; 2016. pp. 223–228. Available at: DOI:10.1109/ICDIM.2015.7381854
10. Barua L., Zou B., Zhou Y. Machine Learning for international freight transportation management: A comprehensive review. Research in Transportation Business and Management. Elsevier; 2020; 34(July 2019): 100453. Available at: DOI:10.1016/j.rtbm.2020.100453
11. Zhang Y., Zhou Z., Fu Y., Zhou J., Yang X., Zhang D. Runway Visual Range Prediction Based on Ensemble Learning. Proceedings 2018 Chinese Automation Congress, CAC 2018. IEEE; 2019; : 3127–3132. Available at: DOI:10.1109/CAC.2018.8623776
12. Chen J., Li M. Chained predictions of flight delay using Machine Learning. AIAA Scitech 2019 Forum. 2019. pp. 1–25. Available at: DOI:10.2514/6.2019-1661

13. Meel P., Singhal M., Tanwar M., Saini N. using Machine Learned Classifiers. 2020; : 71–76.
14. Mukhiya SK., Ahmed U. Hands-On Exploratory Data.
15. Wu Q., Hu M., Ma X., Wang Y., Cong W., Delahaye D. Modelling Flight Delay Propagation in Airport and Airspace Network. IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. 2018; 2018-Novem: 3556–3561. Available at: DOI:10.1109/ITSC.2018.8569657
16. Henriques R., Feiteira I. Predictive modelling: Flight delays and associated factors, Hartsfield-Jackson Atlanta international airport. Procedia Computer Science. Elsevier B.V.; 2018; 138: 638–645. Available at: DOI:10.1016/j.procs.2018.10.085
17. Hariri RH., Fredericks EM., Bowers KM. Uncertainty in data analytics: survey, opportunities, and challenges. Journal of Data. Springer International Publishing; 2019; 6(1). Available at: DOI:10.1186/s40537-019-0206-3
18. Chung SH., Ma HL., Hansen M., Choi TM. Data science and analytics in aviation. Transportation Research Part E: Logistics and Transportation Review. 2020; 134(January). Available at: DOI:10.1016/j.tre.2020.101837
19. Zeng N., Qiu H., Wang Z., Liu W., Zhang H., Li Y. A new switching-delayed-PSO-based optimised SVM algorithm for diagnosis of Alzheimer's disease. Neurocomputing. Elsevier B.V.; 2018; 320: 195–202. Available at: DOI:10.1016/j.neucom.2018.09.001
20. Cichos F., Gustavsson K., Mehlig B., Volpe G. Machine Learning for active matter. Nature Machine Intelligence. Springer US; 2020; 2(2): 94–103. Available at: DOI:10.1038/s42256-020-0146-9
21. Kreger J., Fischer L., Hasler S., Weisswange TH., Bauer-Wersing U. A priori reliability prediction with meta-learning based on context information. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2017. 200–207 p. Available at: DOI:10.1007/978-3-319-68612-7_23
22. Hofmann H., Wickham H., Kafadar K. Letter-Value Plots: Boxplots for Large Data. Journal of Computational and Graphical Statistics. Taylor & Francis; 2017; 26(3): 469–477. Available at: DOI:10.1080/10618600.2017.1305277
23. Hughes D., Correll N. Distributed Machine Learning in Materials that Couple Sensing, Actuation, Computation and Communication. 2016; : 1–80. Available at: <http://arxiv.org/abs/1606.03508>
24. Heer J., Kong N., Agrawala M. Sizing the horizon. 2009; : 1303. Available at: DOI:10.1145/1518701.1518897
25. Sankaranarayanan HB., Vishwanath B V., Rathod V. An exploratory analysis for predicting passenger satisfaction at global hub airports using logistic model trees. Proceedings - 2016 2nd IEEE International Conference on Research in Computational Intelligence and Communication Networks, ICRCICN 2016. IEEE; 2017; : 285–290. Available at: DOI:10.1109/ICRCICN.2016.7813672
26. Efthymiou M., Njoya ET., Lo PL., Papatheodorou A., Randall D. The impact of delays on customers' satisfaction: An empirical analysis of the british airways on-time performance at heathrow airport. Journal of Aerospace Technology and Management. 2019; 11: 1–13. Available at:

- DOI:10.5028/jatm.v11.977
27. Kumar S., Zymbler M. A Machine Learning approach to analyse customer satisfaction from airline tweets. *Journal of Data*. Springer International Publishing; 2019; 6(1): 1–16. Available at: DOI:10.1186/s40537-019-0224-1
 28. Gatersleben MR. Analysis and simulation of passenger flows in an airport terminal. *Proceedings of the 31st conference on Winter simulation: Simulation---a bridge to the future-Volume 2*. 1999. 1999. pp. 1226–1231. Available at: DOI:10.1017/CBO9781107415324.004
 29. Hunter G., Boisvert B., Ramamoorthy K. Advanced National Airspace traffic flow management simulation experiments and validation. *Proceedings - Winter Simulation Conference*. 2007. pp. 1261–1267. Available at: DOI:10.1109/WSC.2007.4419730
 30. Chakrabarty N. A data mining approach to flight arrival delay prediction for American airlines. *IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference*. 2019. pp. 102–107. Available at: DOI:10.1109/IEMECONX.2019.8876970
 31. Reynolds-Feighan AJ., Button KJ. An assessment of the capacity and congestion levels at European airports. *Journal of Air Transport Management*. 1999; 5(3): 113–134. Available at: DOI:10.1016/S0969-6997(99)00006-X
 32. Baspinar B., Ure NK., Koyuncu E., Inalhan G. Analysis of Delay Characteristics of European Air Traffic through a Data-Driven Airport-Centric Queuing Network Model. *IFAC-PapersOnLine*. Elsevier B.V.; 2016; 49(3): 359–364. Available at: DOI:10.1016/j.ifacol.2016.07.060
 33. European Union Aviation Safety Agency. *Artificial Intelligence Roadmap: A Human-Centric Approach to AI in Aviation*. 2020; (February): 1–33. Available at: <https://www.easa.europa.eu/newsroom-and-events/news/easa-artificial-intelligence-roadmap-10-published>
 34. Cook Di., Lee EK., Majumder M. Data Visualization and Statistical Graphics in Data Analysis. *Annual Review of Statistics and Its Application*. 2016; 3: 133–159. Available at: DOI:10.1146/annurev-statistics-041715-033420
 35. Akerkar R. Analytics on big aviation data: Turning data into insights. *International Journal of Computer Science and Applications*. 2014; 11(3): 116–127.
 36. Khanmohammadi S., Chou CA., Lewis HW., Elias D. A systems approach for scheduling aircraft landings in JFK airport. *IEEE International Conference on Fuzzy Systems*. IEEE; 2014. pp. 1578–1585. Available at: DOI:10.1109/FUZZ-IEEE.2014.6891588
 37. Liu, Xia, Fang Wan, Lei Chen, Zhao Qiu and MC. Research on traffic passenger volume prediction of Sanya city based on ARIMA and grey markov models. *4th International Conference of Pioneering Computer Scientists, Engineers and Educators Part II*. Springer, Singapore; 2018. 69–74 p. Available at: DOI:10.30748/soi.2018.153.08
 38. Hou S., Gong Z., Nie Q., Xiao Q., Tan Y. A hybrid predictive model for high-frequency and multi-periodic data in call center of online travel agency. *Proceedings - 2018 IEEE International Conference on Data, Data 2018*. IEEE; 2019; d: 3128–3134. Available at:

- DOI:10.1109/BigData.2018.8621977
39. Reddy Edla D., Lingras P., Venkatanareshbabu K., SpringerLink. Advances in Machine Learning and data science: recent achievements and research directives. Springer Singapore; 2018. 377 p.
 40. Liu Y., Jia G., Tao X., Xu X., Dou W. A stop planning method over big traffic data for airport shuttle bus. Proceedings - 4th IEEE International Conference on Data and Cloud Computing, BDCloud 2014 with the 7th IEEE International Conference on Social Computing and Networking, SocialCom 2014 and the 4th International Conference on Sustainable Computing and C. IEEE; 2014; : 63–70. Available at: DOI:10.1109/BDCloud.2014.21
 41. AnHai D., Alon H., Zachary I. Principles of Data Integration. Principles of Data Integration. 2012; Available at: DOI:10.1016/C2011-0-06130-6
 42. Squire M. Mastering Data Mining with Python – Find patterns hidden in your data. Packt Publishing. 2016. 269 p. Available at: www.packtpub.com
 43. Gupta D. Applied analytics through case studies using SAS and R: Implementing predictive models and Machine Learning techniques. Applied Analytics through Case Studies Using SAS and R: Implementing Predictive Models and Machine Learning Techniques. 2018. 1–404 p. Available at: DOI:10.1007/978-1-4842-3525-6
 44. Dhanawade R., Deo M., Khanna N., Deolekar R V. Analysing factors influencing flight delay prediction. Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development, INDIACom 2019. 2019; : 1003–1007.
 45. Saxe AM., McClelland JL., Ganguli S. Exact solutions to the non-linear dynamics of learning in deep linear neural networks. 2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings. 2014; : 1–22.
 46. Tang Q., Yang M., Yang Y. ST-LSTM: A Deep Learning Approach Combined Spatio-Temporal Features for Short-Term Forecast in Rail Transit. Journal of Advanced Transportation. 2019; 2019. Available at: DOI:10.1155/2019/8392592
 47. Cheng F., Baszczewski B., Guldin J. A hybrid optimisation-simulation approach for itinerary generation. Proceedings - Winter Simulation Conference. IEEE; 2015. pp. 1885–1896. Available at: DOI:10.1109/WSC.2014.7020036
 48. Hu C., Wei H., B MC. Enhancing Airlines Delay Prediction by Implementing Classification Based Deep Learning Algorithms. Springer International Publishing; 2019. 563–581 p. Available at: DOI:10.1007/978-3-030-19063-7
 49. Najafabadi MM., Villanustre F., Khoshgoftaar TM., Seliya N., Wald R., Muharemagic E. Deep Learning applications and challenges in data analytics. Journal of Data. 2015; 2(1): 1–21. Available at: DOI:10.1186/s40537-014-0007-7
 50. Yiu C. The data opportunity. Policy Exchange. 2012. 34 p. Available at: [http://www.policyexchange.org.uk/images/publications/the data opportunity.pdf](http://www.policyexchange.org.uk/images/publications/the_data_opportunity.pdf)
 51. Cutajar K., Bonilla E V., Michiardi P., Filippone M. Random feature

- expansions for Deep Gaussian Processes. 34th International Conference on Machine Learning, ICML 2017. 2017; 2: 1467–1482.
52. Granitto PM., Furlanello C., Biasioli F., Gasperi F. Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products. *Chemometrics and Intelligent Laboratory Systems*. 2006; 83(2): 83–90. Available at: DOI:10.1016/j.chemolab.2006.01.007
 53. Liang H., Sun X., Sun Y., Gao Y. Text feature extraction based on Deep Learning: a review. *Eurasip Journal on Wireless Communications and Networking*. EURASIP Journal on Wireless Communications and Networking; 2017; 2017(1): 1–12. Available at: DOI:10.1186/s13638-017-0993-1
 54. Hensman J., Durrande N., Solin A. Variational fourier features for Gaussian processes. *Journal of Machine Learning Research*. 2018; 18: 1–52.
 55. Sternberg A., Carvalho D., Murta L., Soares J., Ogasawara E. An analysis of Brazilian flight delays based on frequent patterns. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2016; 95: 282–298. Available at: DOI:10.1016/j.tre.2016.09.013
 56. Abdel-Aty M., Lee C., Bai Y., Li X., Michalak M. Detecting periodic patterns of arrival delay. *Journal of Air Transport Management*. 2007; 13(6): 355–361. Available at: DOI:10.1016/j.jairtraman.2007.06.002
 57. Surya Teja KP., Reddy V., Subhani S. Flight delay prediction using Machine Learning algorithm xgboost. *Journal of Advanced Research in Dynamical and Control Systems*. 2019; 11(5): 379–388.
 58. Klein A., Kavoussi S., Lee RS. Weather forecast accuracy: Study of impact on airport capacity and estimation of avoidable costs. *Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009*. 2009. pp. 12–21.
 59. Gopalakrishnan K., Balakrishnan H. A comparative analysis of models for predicting delays in air traffic networks. *12th USA/Europe Air Traffic Management R and D Seminar*. 2017;
 60. Xu N., Donohue G., Laskey KB., Chen CH. Estimation of delay propagation in the national aviation system using Bayesian networks. *Proceedings of the 6th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2005*. 2005. pp. 353–363.
 61. Kotegawa T., De Laurentis D., Noonan K., Post J. Impact of commercial airline network evolution on the US air transportation system. *Proceedings of the 9th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2011*. 2011. pp. 572–580.
 62. Liu X., Li L., Liu X., Zhang T., Rong X., Yang L., et al. Field investigation on characteristics of passenger flow in a Chinese hub airport terminal. *Building and Environment*. 2018; 133: 51–61. Available at: DOI:10.1016/j.buildenv.2018.02.009
 63. Liu H., Lin Y., Chen Z., Guo D., Zhang J., Jing H. Research on the Air Traffic Flow Prediction Using a Deep Learning Approach. *IEEE Access*. 2019; 7: 148019–148030. Available at: DOI:10.1109/ACCESS.2019.2945821
 64. Perin C., Vernier F. Progressive Horizon Graphs: Improving Small Multiples Visualisation of Time Series. *VisWeek 2012 Electronic*

- Conference Proceedings (2012). 2012; (2012).
65. Gui G., Liu F., Sun J., Yang J., Zhou Z., Zhao D. Flight delay prediction based on aviation data and Machine Learning. *IEEE Transactions on Vehicular Technology*. 2020; 69(1): 140–150. Available at: DOI:10.1109/TVT.2019.2954094
 66. Chandramouleeswaran KR., Krzemien D., Burns K., Tran HT. Machine Learning prediction of airport delays in the US air transportation network. 2018 Aviation Technology, Integration, and Operations Conference. 2018; : 1–10. Available at: DOI:10.2514/6.2018-3672
 67. Pfeil DM., Balakrishnan H., Zou B., Hansen M., Zonglei L., Jiandong W., et al. A Recursion-based approach to simulating Airline schedule robustness. *Journal of Air Transport Management*. Elsevier Ltd; 2008; 2(1): 1–11. Available at: DOI:10.1016/j.jairtraman.2006.07.004
 68. Popovic V., Kraal B., Philip K. Towards airport passenger experience models. In *Proceedings of the 7th International Conference on Design and Emotion 2010*. IIT Institute of Design: IIT Institute of Design; 2010. pp. 1–11.

Chapter 4: A Deep Feedforward Neural Network and Shallow Architectures Effectiveness Comparison: Flight Delays Classification Perspective

Flight delays have negatively impacted passengers, airlines and airports' socio-economics state, resulting in substantial economic losses. Hence, it has become necessary to predict their occurrences correctly in decision-making because it is essential for the aviation industry's effective management. Developing accurate flight delay classification models depends mostly on the air transportation system complexity and airport infrastructure, which may be a region-specific issue. However, no specific prediction or classification model can simultaneously handle all airline's and airports' characteristics. Hence, the need to further develop and compare predictive models for the future aviation decision system cannot be over-emphasised. In this research, flight on-time data records from the United States Bureau of Transportation Statistics were employed to evaluate the performances of Deep Feedforward Neural Network, Neural Network, and Support Vector Machine models on a binary classification problem. The research revealed that the models achieved different accuracies of flight delay classifications. In the initial experiment, the Support Vector Machine had the worst average accuracy than Neural Network and Deep Feedforward Neural Network. The Deep Feedforward Neural Network outperformed Support Vector Machines and Neural Networks with the best average percentage accuracies. Further investigating the Deep Feedforward Neural Network architecture on different parameters against itself suggests training a Deep Feedforward Neural Network algorithm. Regardless of data training size, the classification accuracy peaks. We examine which number of epochs works best in our flight delay classification settings for the Deep Feedforward Neural Network. Our experiment results demonstrate that many epochs affect the model's convergence rate, unlike when we increase hidden layers. It does not ensure better or higher accuracy in a binary classification of flight delays. Finally, we recommended further studies on the applicability of Deep Feedforward Neural Networks in flight delay prediction with specific case studies of either airlines or airports to check the impact on the model's performance.

4.1 Introduction

Human social life in modern times has increasingly become dependent on the aviation industry, and accurate classification of flight delays is profitable but difficult due to uncertainties and failures that might occur within the processes. Despite the possibility of using land or sea transport for travel, the efficiency and reliability of air transportation have made it a better option for most passengers [1]. Flight delays have increased due to the rapid increase in air traffic and have become a prominent problem in the aviation industry [1]. The Bureau of Transportation Statistics (BTS) report states that one out of four flights arrived at a destination later than 15 minutes [2]. Also, the annual cost of air transportation delays totals over \$30 billion, posing a challenge to the design and development of the Next-generation Air Transportation Decision Systems (NATDS) [3]. It motivates the accurate and practical prediction of flight delays, especially on an individual or group basis.

Flight delay propagation is usually caused by the late arrival of a certain flight which may cause the same aircraft to depart late and is beyond the limited airspace capacity delays [4]. Airlines fly their aircraft on a daily schedule itinerary requiring a visit to other airports; aircraft arriving late in the day significantly impacts the delay performance after that [5]. For instance, an aircraft delayed for one hour from the departing airport will certainly delay arriving at the next airport; the lateness in arriving will affect the aircraft's subsequent departure, leading to subsequent lateness in aircraft arrivals [6]. The United States (US) National Airspace System (NAS) is subject to delay propagation, including but not limited to many resources connected, such as the crew, passenger, aircraft and gate space. The slack time between departure and arrival is forced to reduce in the NATSDS due to increasing air traffic demands, making the NAS suffer more network delays. Therefore, modelling flight delays is key to successful and accurate flight delay predictions or classifications.

For several years, flight delays have been one of the major problems costing the airline industry. Several reasons can cause flight delays ranging from weather conditions (strong winds, winter storms, etc.), late-arriving aircraft, NAS (heavy traffic volume, air traffic control), security, and air carrier (aircraft cleaning,

baggage loading, etc.) [7]. Additional operation expenses are incurred when a flight is delayed, mostly due to increased spending on crews, maintenance, and fuel consumption. This inconveniences passengers and negatively affects the overall economy [8]. Flight delay studies have been important research areas with much attraction, leading to the employment of data analytics and Machine Learning methods to predict airline flight delays [9]. If airlines could predict flight delays, a proactive decision could be taken to effectively manage disruptions and minimise last-minute flight rescheduling, thus improving travel experiences. Approximately 20% of airline flights are delayed or cancelled yearly, mainly due to several causes [10]. These delays significantly incur a cost to both passengers and airlines. For example, in 2007, the US economy was affected by a cost of \$32.9 billion [10]. Over half of that cost accounted for passengers' claims to the airlines due to incurred delays.

However, there is no superior or optimal solution to the problem of flight-delay classifications and predictions to date [11]. Several studies attempt to find the most suitable solutions to flight delays using Deep Learning (DL) methods. These methods were applied to datasets ranging from weather-caused to non-weather-caused delays. The use of weather data to predict Beijing Capital International Airport departure capacity was carried out by Xiangmin in [11]. Both Choi et al. in [12] and Belcastro et al. in [13] applied weather and flight data for predicting delays. In [14], weather forecast information of origin and destination airports were utilised for prediction. In contrast, in [15], weather characteristics such as wind direction, speed rate, snow depth, precipitation, and visibility were used to predict flight delays. The author in [16] proposed a Machine Learning model focusing on weather-induced delays on individual flights. Binary classification accuracy for a specific individual origin-destination pair can be improved by fine-tuning parameters such as activation function, the number of hidden layers, and feature selection. Overall, most of the previous studies have their limitation in feature selection techniques. This work aims to implement a Deep Learning classifier for non-weather caused flight delays and compare the performance against other popular shallow architecture Machine Learning methods.

We can summarise the contribution of this paper in two points. Firstly, we introduce a new algorithm that supports classifying both departure and arrival delays that combines a feature selection method known as filter-based feature selection from [17] for a multi-label classification algorithm Deep Feedforward Neural Network (DNN). Based on the BTS flight on-time dataset, we applied a method to select a feature for optimal model training for high classification accuracy. Secondly, according to the International Air Transport Association (IATA), one of the key causes of airline delays is the delayed arrival of an aircraft. In our model, we have captured this relationship by considering the knock-on effects. Also, we made a performance comparison for three models, DNN and shallow architectures known as Support Vector Machine (SVM) and one-layer Neural Network (NN), on a flight delay classification task.

The rest of the paper is organised as follows: Section 4.2 discusses related work and main concepts. Section 4.3 discusses the methods implemented. Section 4.4 discusses data and performance metrics for evaluating and assessing the model quality. Finally, Section 4.5 concludes and presents future directions.

4.2 Related Work

Flight delays have been causing numerous problems for both passengers and the aviation industry [7]. Researchers have used collective efforts in the quest to address this problem by providing solutions through experimental findings [16]. Despite the huge successes in addressing this problem, there is no optimal solution. Therefore, there is a need for more research and investigation of the issue [18]. The authors in [19] developed a methodology for flight schedules. A clustering approach was employed to group airports into clusters, which were used as input into the flight scheduling system. However, the author suggests that the system can be improved using more sophisticated methods in the future. Fei [20] performed a factor analysis of factual data from irregular flights, which combines a Gaussian Mixture Model and Expectation Maximisation (GMM-EM) algorithm. The model was tested on the Air Traffic Management Bureau (ATMB). It was discovered that the upstream delay positively affects the quality of downstream delay prediction. The authors in [21] applied Deep Recurrent Neural

Network (DRNN) in discussing the architecture of Long Short-Term Memory (LSTM) and RNN networks. They explored the benefit of stacking the neural networks and ways of making the architectures deeper using RNN. The use of deep LSTM-RNN architecture improves the accuracy of the prediction model. Also, Kai-Quan Cai [22] investigated the problem of multi-objective air traffic network flow optimisation (MATNFO). A method for systematic searches of flight routes and time of departure-arrival was proposed. A sequential time slot for flight routes searched to avoid computational overhead was implemented. The result of the experiment shows that their approach performs better than SVM.

In [16], the authors developed a model with two phases of prediction as follows: (i) Delay on departure; and (ii) Delay on arrival. This was for effective prediction of departure and arrival delays using a schedule of flight and weather forecast features. A binary classification to forecast delay occurrence and regression to find delay value in minutes were implemented. Gradient Boosting Classifier performed best during the classification, while the Extra-Tree regressor performed best in the regression task. The departure delay prediction has a higher error rate than the arrival delay prediction. In [23], the author developed a unimodal model for predicting time delays. Neural networks and the binomial model with trees were implemented and compared. The author realises arrival delay and ground operation time as the most significant indicators in predicting departure delay. In [24], the author implemented the Gradient Boost model for flight delay prediction. The highest coefficient of determination of 92.3185% for arrival and 94.8523% for departure was achieved. However, the model was limited to only 70 airports. Also, the author in [25] implemented a Neural Network and Deep Belief Network for flight delay prediction. 92% and 77% accuracy were achieved for Neural Network with three hidden layers of neurons and Deep Believe Network with four hidden layer neurons.

In this work, we conducted a thorough performance comparison of three popular Machine Learning methods known as DNN, NN and SVM in a flight delay classification problem task. Also, to further investigate the performance of DNN architecture with different parameter settings.

Problem Definition

Let flight F_1 be a tuple $(B_o, B_d, t_{sd}, t_{ad}, t_{sa}, t_{aa})$ where B_o represents the origin airport, B_d represents the destination airport, t_{sd} represents the scheduled departure time, t_{ad} represents the actual departure time, t_{sa} represents the scheduled arrival time to the gate, while t_{aa} represents the actual arrival time to the gate. All dates, hours and minutes are included in time. As stated earlier, this research is for the classification of non-weather-induced flight delays consisting of both departure and arrival delays. The classification considers information related to both departure and arrival flights (origin airport, destination airport, departure scheduled time and arrival schedule time) according to the schedule. The delay classification of any flight F_1 departing from airport B_o at a time t_{sd} and arriving at an airport B_d , at a time t_{sa} is an estimated arrival delay $AD(F_1)$, while departure delay $DD(F_1)$ is when a flight departs the origin airport later than the scheduled time. If the delay prediction is below a certain threshold T_1 , then it is classified as an on-time flight. Else, it is classified as being delayed flight.

4.3 Methods

In this section, we describe the ML methods employed in our research. These are among the most popularly used ML methods in the literature for classification problems [25, 42]. Table 4-1 shows the parameters and levels used in training the three methods (SVM, NN and DNN) compared in this research.

4.3.1 Support Vector Machine

Support Vector Machines belong to general kernel methods [26, 27, 28], which depend on data only through dot products. A kernel function n is used to replace the dot product by computing a high dimensional feature space dot product of the data. It has two major advantages: the ability to generate non-linear decisions with linear classifiers and the ability of the kernel to allow classifiers to be applied to no obvious dimensional vector space representation data. In modelling SVM, a hyperplane is constructed to form a decision boundary creating a margin between positive and negative classes [30, 31]. The SVM model learns the classification into two classes, given a training set with $x_i \in \mathbb{R}^d$ of input vectors with corresponding $y_i \in \{+1, -1\}$ labels.

A two-class problem of flight delays, having a label of 1 (there exists a delay) and 0 (there exists no delay). Let x be a vector with elements x_i ; in our case, the input features containing i^{th} vector in a dataset composed of n labelled examples (x_i, y_i) where y_i is the label and x_i denotes the input variables. A linear classifier is the dot product between two vectors, the scalar or inner product. Equation (4-1) defines a linear classifier with a linear discriminant function:

$$f(x) = \mathbf{w}^T x + b \quad (4-1)$$

where x is the input variable, w is the weighted vector, and b is the bias. For the case of $b = 0$, this means that for the set of points x_i such that $w^T = 0$ are points that go through the origin and perpendicular to w , making a hyperplane that divides the space according to the sign of discriminant function $f(x)$ into two. The classifier's decision boundary is the boundary region classified as positive or negative. A classifier that has a linear decision boundary is referred to as a linear classifier; otherwise, if dependent on non-linear data is known as a non-linear classifier [31, 32]

The type of kernel function, the degree of kernel function (d : polynomial kernel; γ : for the radial base kernel), and regularisation constant c form the parameter of an SVM. We used the approach proposed in [34] to determine efficient parameters. In our experiment, we applied five levels of d , and ten levels of γ , while both four and five levels of c were tested in the parameter settings. We then selected the SVM model's best polynomial and radial basis compared to the NN and DNN models.

4.3.2 Neural Network

Neural Network is a non-linear prediction method and one of the popular methods applied in classification problems such as stock index prediction, fraud detection etc. [34, 35–57]. It is referred to as Multilayer perceptron (MLP) or feedforward NN. The dense interconnected layers help the inputs to activate the neurons. In this study, we employ a one-layer feedforward NN. The input for the model is as simple as the same in DNN. We use a single-layer output neuron and a transfer function, known as a log function, which produces a continuous output value between 0 and 1. To determine the class of delay, we use a threshold of

0.5. If the output is greater than or equal to 0.5, this will result in a delay classification. Using the threshold of 0.5 was the best in producing the most accurate classification results. Additionally, in line with the work presented in [33, 34, 35, 36, 48], a sigmoid function was employed as a transfer function for the hidden layers because it was set with the most accurate classification results in previous research. Gradient Descent was used at each epoch to adjust the weights to attain a global minimum.

The NN has several numbers of hidden layers neurons (n), the learning rate value (lr), the number of epochs (ep), and the momentum constant (mc). To determine the most appropriate parameters, we tested ten different values of n , nine different values of mc and ten different values of ep . Then, we selected the best result for comparison with the DNN and SVM. We found that the learning rate lr for the top-performing settings in the model was within the interval [0.1, 0.2].

4.3.3 Deep Feedforward Neural Network

A Deep Feedforward Neural Network consists of at least an input layer, two or more hidden layers and an output layer. Several functions are attached to DNNs [29, 40]. For instance, We might have five functions $f^{(1)}$, $f^{(2)}$, $f^{(3)}$, $f^{(4)}$ and $f^{(5)}$, all in a chain connection forming a function $f(x) \approx (f^{(5)}, f^{(4)}, f^{(3)}, f^{(2)}, f^{(1)}(x))$. While in this case, $f^{(1)}$ is known as the first layer, $f^{(2)}$ is known as the second layer, and it continues until the final layer, which is called the output layer. The output layer is the same for either classification or regression tasks matching the output space. The overall length of the network of the chain gives the model depth. The term Deep Learning comes from this concept. We employ classical feedforward architecture as shown in Equation (4-2). The neurons in previous layers are fully connected with all neurons in $1 + l$ subsequent layer with a weighted directed edge. Also, a bias unit connected to all non-output layers of the network serves as the subsequent layer's activation threshold. Each weighted combination of y of the n_l Each neuron receives the neuron outputs in the previous layer l as input.

$$y = \sum_{i=1}^{n_l} x_i w_i + b \quad (4-2)$$

where w_i represents the weight of the output, x_i and b represent the input and bias, respectively. The activation function a transforms the weighted combination γ so that output signal $g(\gamma)$ is relayed on subsequent layer neurons, represented as in Equation (4-3).

$$a = g(\gamma) = g\left(\sum_{i=1}^{n_l} x_i w_i + b\right) \quad (4-3)$$

where a is the activation function, $g(\gamma)$ output signal, w_i are the weights, and b is the bias that we adjusted during the training process of the network manually based on some rules until an optimal result is obtained. Following the work proposed in [42], we used the Sigmoid activation function on all the other layers except the output layer, where a SoftMax activation function was applied to constrict real value into $[0, 1]$ based on our target output. These activation functions can be represented in Equations (4-4) and (4-5).

$$a = g(\gamma) = \frac{1}{1 + e^{-\gamma}} \quad (4-4)$$

$$a = g(\gamma) = \frac{e^{\gamma}}{\sum_{l=0}^k e^{\gamma_l}} \quad (4-5)$$

where $e^{-\gamma}$, e^{γ} and e^{γ_l} are the exponential decay of input values and k, l are the initial value and terminating value during training. But the sigmoid activation function is a differentiable function as shown in Equation (4-6) where $g'(\gamma)$ are the decomposed function and all other parameters maintain their usual meanings.

$$a' = g'(\gamma) = \frac{e^{-\gamma}}{(1 + e^{-\gamma})^2} = \frac{1}{1 + e^{-\gamma}} \frac{e^{-\gamma}}{(1 + e^{-\gamma})} = \frac{1}{1 + e^{-\gamma}} \left(1 - \frac{e^{-\gamma}}{(1 + e^{-\gamma})}\right) = g(\gamma)(1 - g(\gamma)) \quad (4-6)$$

Let W be the collection of $\cup_{l=1}^{L-1} w_l$, where w_l is the weighted matrix that connects two consecutive layers l and $l + 1$ for a network with L number of layers. Similarly, assume B to be the collection $\cup_{l=1}^{L-1} b_l$ with c_1 column vector of biases of l layer. The output of the entire DNN network is fully determined by the collection of W and B as shown in Equation (4-7). We adapt the weight accordingly to minimise the error in training on the data. Specifically, the objective is minimising some loss function $L(W, B|j)$ for j set of training examples. We are solving a

classification problem related to flight delays, and the cross-entropy loss function in Equation (4-7) is applied.

$$L(W, B|j) = -\sum_{y \in 0} (\ln(o_y^{(j)} t_y^{(j)} + \ln(1 - o_y^{(j)})(1 - t_y^{(j)})) \quad (4-7)$$

where the output units are represented by y , o represents the output layers, $o_y^{(j)}$ is the predicted class, and $t_y^{(j)}$ is the actual output class, for instance, j . We used Stochastic Gradient Descent (SGD) to minimise the loss function. Also, we employed Backward Propagation to calculate the Gradient loss function $L(W, B|j)$. Since the input variables are simple (e.g., seven variables), we used a straightforward architecture of deep feedforward neural network as shown in Figure 4-1, where the input layer, hidden layers and output layer. Figure 4-1 shows that the interconnections between hidden neurons can be complicated during training because of the random value generation for the weight and bias at each training epoch. This is one of the interesting aspects of dealing with deep neural networks.

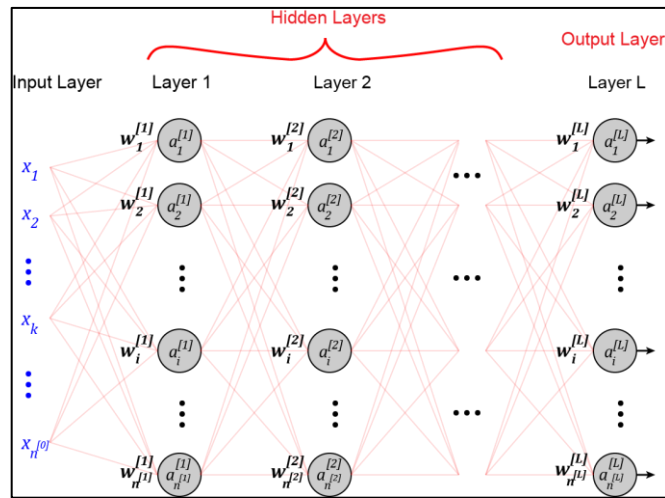


Figure 4-1 Architecture of Deep Feedforward Neural Network [43]

Table 4-1 shows the parameter settings used for each method during the model training. These parameters are specific to each of the model's architecture and are fine-tuned to optimise the expected output of the models. This is important in every Machine Learning task because the parameters have the overall control of any model before applying a special optimisation algorithm if needed. Table 4-1 shows that the three Machine Learning methods SVM, NN

and DNN implemented in this paper have their architectural characteristics for both the linear learner (SVM) and the non-linear learners (NN, DNN). DNN is an extension of the NN in trying to improve the learnability of the model with more layers and fine-tuning parameters.

Table 4-1 Parameter settings for SVM, NN and DNN

S/No.	Method	Levels
1	SVM	polynomials
	Kernel function degree (d)	1, 2, 3, ..., 5
	Kernel function Gamma (γ)	0, 0.1, 0.2, ..., 5.0
	Regularization (c)	1, 10, 50, 100
2	NN	
	Epochs (ep)	1, 2, 3, ..., 10
	Learning rate (lr)	0.1, 0.2
	Momentum constant (mc)	0.1, 0.2, 0.3, ..., 0.9
	Number of neurons (n)	10, 20, 30, ..., 100
3	DNN	
	Epochs (ep)	1, 2, 3, ..., 10
	Learning rate (lr)	0.1, 0.2
	Momentum constant (mc)	0.1, 0.2, 0.3, ..., 0.9
	Size of Mini Batch	5, 6, 7, ..., 15
	Hidden layers sizes	2, 3, 4, ..., 10
	Activation functions	Sigmoid/SoftMax
	Number of neurons (n)	10, 20, 30, ..., 100

4.4 Discussion of Results

In this paper, we evaluated individual flight delays for each one of the methods discussed earlier. The performance of SVM, NN and DNN was measured in the experiment in classifying flight delays to find out which of the methods performed best in our case. We also explored the differences in data handling of these methods during model training. To find out this, we conduct a binary classification problem using the three methods to classify individual flight delays into a class of all flights that depart or arrive 10 minutes after the scheduled period to be considered delayed, otherwise not delayed.

4.4.1 Experimental Setup

We conducted repeated experiments to clarify the factors influencing the predictability of the simple ML classifiers (SVM, NN). Figure 4-2 shows the steps in the general experiment. The implementation was performed using Python programming language and relevant libraries embedded in Jupyter Notebook version 6.0.1 [40] on a Graphics Processing Unit (GPU). Detailed information about the software environment is provided in Table 4-2.

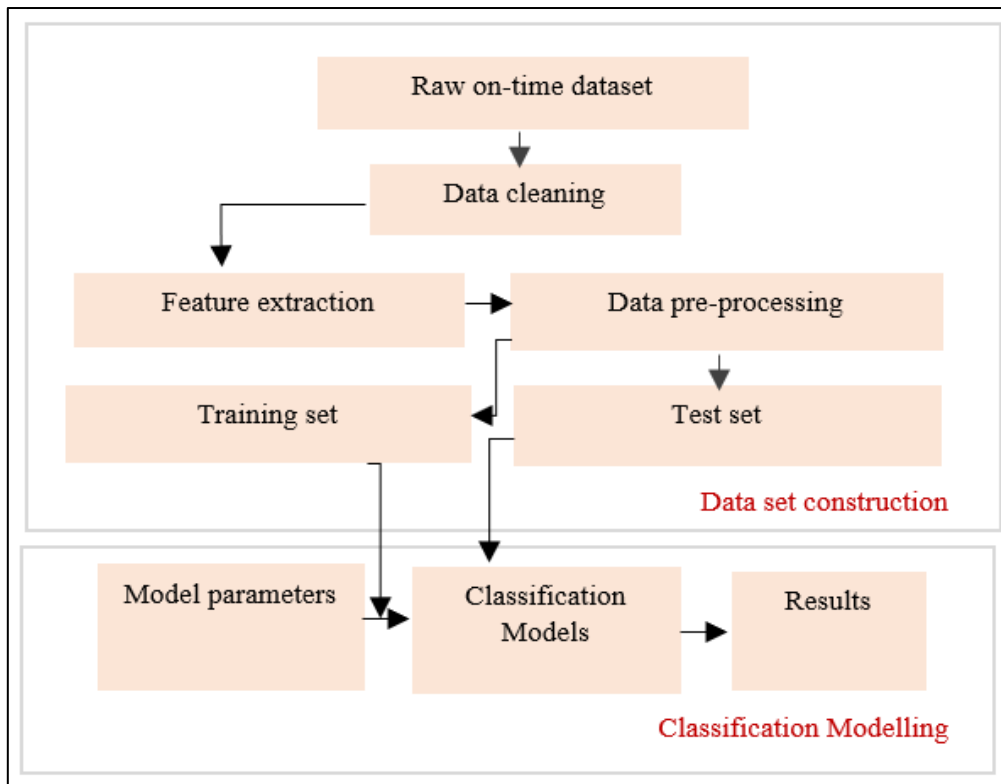


Figure 4-2 An Overview of the Experiment Steps

Table 4-2 Software environment information

S/No	Name of library	Version
1	keras	2.3.1
2	numpy	1.19.5
3	pandas	0.25.1
4	itertools	0.25.1
5	matplotlib	3.1.1
6	sklearn	0.22.2.post1

7	tensorboard	2.1.1
8	tensorflow	2.1.0
9	tensorflow-gpu	2.0.0
10	ipython	7.8.0

Dataset Description

Table 4-3 contains the factors that influence the predictability of the flight delay classification. A real-world dataset for flight on-time data from the US BTS [14] was used for the three models evaluated in this research. The flight on-time dataset used covers a time from January to December 2018. More than 100 arrival and departure airports are covered in the data, with more than 150,000 training samples. The weather feature was not considered because of its low impact on predictive accuracy improvement [12]. A preliminary dataset with a high probability of flight departure and arrival delay was selected. The original dataset contains 21 factors. However, some factors were removed due to their irrelevance in our targeted output. The summary of the employed factors is presented in Table 4-3.

Table 4-3 Dataset factors information

S/No.	Factor	Unit	Detail
1	Flight departure and arrival airport	Text	3 letters code eg. LHR
2	Scheduled flight departure and arrival time	Hours and Minutes	In four digits formats eg. 1456
3	Flight number	Numeric	Flights number eg. 1454
4	Actual departure and arrival time	Hours and Minutes	In four digits formats eg. 1456
5	Unique carrier	Numeric	3 letters code eg. PS
6	Scheduled and actual elapsed time	Numeric	In two digits formats eg. 94
7	Departure and arrival delay	Numeric	Number of flights eg. 14

Dataset Cleaning

We deleted incomplete data. The data may not have a complete record due to message transmission problems. Also, obvious abnormal features were

deleted in the quest to improve the accuracy of the model classification in the training set:

- i. Duplicate data in the departure and arrival airports.
- ii. Data with a very long delay time.

Feature Extraction

The feature extracted for this research is from the factors affecting the classification of flight delays using Algorithm 4-1. Flight plan formulation is subject to many departments, such as airlines and airports, which all parties are interested in, even with their restricted factors. The features are extracted based on their impact on either the departure or arrival flight delay before applying some pre-processing techniques to fine-tune the input dataset for the ML model training.

Algorithm 4-1: Feature Extraction Approach

```
1: Procedure:
2:   begin
3:   Read data
4:   for 1 to N features do
5:     select the best feature subset
6:     apply learning algorithm
7:     evaluate selected feature subset
8:   end for
9:   return final selected feature subset
10: end
```

Dataset Pre-processing

We had to pre-process the data differently because SVM works with numerical values, and NN/DNN works with text and numerical data input values. We define the delay time as the differences between the scheduled time and actual departure time, and in turn, we then calculate the median delay time. We consider the delays of scheduled arrival and departure times according to the hour of the day and date of the month. Finally, the model considers the arrival and departure airports.

4.4.2 Results

We conducted a binary classification on individual flight departure and arrival on-time records; the comparison of the results of the three ML approaches evaluated for the flight delay classification problem is discussed in this section.

Performance Metrics

To evaluate the quality of our results, we are using several metrics. The most significant metric to measure the performance of classification Machine Learning models is that of *Accuracy* [32, 40]. *Precision* and *recall* are important metrics because they help further evaluate a model's effectiveness. These metrics use the notions of True positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). More precisely, these metrics are defined as follows:

Accuracy (AC) measures the fraction of predictions that the model classified correctly [44]. Formally, accuracy can be defined as shown in Equation (4-8):

$$\text{Accuracy (AC)} = \frac{TP + TN}{TP + FP + TN + FN} \quad (4-8)$$

Precision (P) measures the exactness of a classifier as the truly classified percentage of positives. Is the weighted average of negative and positive precision [44]. Therefore, high precision means less False Positive and a good model. As expressed in Equations (4-9) and (4-10), respectively.

$$\text{Precision}_{\text{positive}} (P_p) = \frac{TP}{TP + FP} \quad (4-9)$$

$$\text{Precision}_{\text{negative}} (P_n) = \frac{TN}{TN + FN} \quad (4-10)$$

Recall (R) measures the completeness of a classifier as the truly classified percentage of actual positives [44]. Is the weighted average of negative and positive recall. Therefore, high recall means less False Negative and a good model. As expressed in Equations (4-11) and (4-12), respectively.

$$\text{Recall}_{\text{positive}} (R_p) = \frac{TP}{TP + FN} \quad (4-11)$$

$$\text{Recall}_{\text{negative}} (R_n) = \frac{TN}{TN + FP} \quad (4-12)$$

Model Evaluation

In this research, we implemented and evaluated SVM, NN and DNN to compare their performance on flight delay classification modelling tasks. The results in terms of the earlier defined metrics are presented in Table 4-4.

Table 4-4 Performance comparison of SVM, NN and DNN on flight delays prediction using BTS dataset

S/No.	Model	Precision	Recall	Accuracy
1	SVM	0.6695	0.8619	0.6618
2	NN	0.7206	0.8990	0.7432
3	DNN	0.9070	0.9003	0.9001

In our experiment, the average recall, precision, and accuracy of the SVM are 86.19%, 66.95% and 66.18%. The measures for the NN are 89.90%, 72.06% and 74.32%, respectively, while for the DNN, the same measures are 90.03%, 90.70%, and 90.01%. Therefore, the DNN model outperformed the SVM and NN for all metrics. As a non-parametric model, SVM performs worse than the parametric models NN and DNN.

In our investigation, we also wanted to find out if there are any advantages to having a deep neural network as opposed to other shallow models in flight delay binary classification when using DNN. The experiment shows a huge improvement with a deep neural network in the results, with at least a 10% improvement in recall, 20% in precision, and 20% inaccuracy compared to the NN. This implies that in different classification problems, there can be varied results for a method having many parameters to be fitted during training than other shallow methods.

In our study, we investigated the accuracy performance of the DNN based on different parameter levels during training, validation and testing. The results of this study are presented in Table 4-5. We investigated the characteristics at different levels of the model building with the same number of features as inputs. Varying the number of hidden layers between 2 to 6 with an epoch of 100 to 500, we observed that when we used 2 hidden layers with 5 neurons and 150 epochs, the accuracy was 80.00%. However, interestingly, when there was an increase in the number of hidden layers, neurons and epochs, the testing accuracy

increased to a maximum of 99.82%. Therefore, there were several experiments with different setups to check the ability of the classification of the models to perform better. Surprisingly, the accuracy of the experiments was not too different in training, testing and validation with the same parameters combination. But increasing the number of hidden layers does not significantly impact accuracy. Therefore, we can conclude that increasing the number of hidden layers does not ensure better or higher accuracy. However, our experiment achieved a 91.00% accuracy in classifying flight delays, which significantly improved compared to a similar experiment presented in [25], where the accuracy was 76.76% with a Deep Believe Network (DBN) using a sigmoid function. Our accuracy is also better when compared with the results presented in [45]. Several reasons for this improvement could be feature availability, data quality or settings of the algorithm's parameters. Summarily, even when fewer features are applied to DNN, the results are better than the other shallow ML algorithms (SVM, NN) implemented in this paper.

Table 4-5 Performance comparison of DNN on flight delays prediction using BTS dataset with different settings

S/No	Number of hidden layers	Number of neurons in hidden layers	Epoch	Accuracy		
				Training	Validation	Testing
1	2	5	150	0.8201	0.8221	0.8101
2	6	5	100	0.8267	0.8593	0.8213
3	3	50	150	0.8812	0.8991	0.8798
4	4	25	500	0.9701	0.9832	0.9982
5	5	13	200	0.9629	0.9734	0.9945

We then further investigated the DNN model's performance on the same 500 epochs with different parameters, as seen in Table 4-6. We observed that the model performance when we used 2 hidden layers with 50 neurons was better regarding training and validation accuracy, with 98.12% and 98.94%, respectively. However, the best performance settings are observed on the testing when we used 4 hidden layers and 20 neurons with 99.85% accuracy. Finally, we

observed that the model performance dropped in training, validation and testing with fewer hidden layers and neurons.

Table 4-6 Performance comparison of DNN on flight delays prediction using BTS dataset with different settings and 500 epochs

S/No	Number of hidden layers	Number of neurons in hidden layers	Accuracy		
			Training	Validation	Testing
1	2	5	0.9201	0.9221	0.9011
2	6	5	0.9265	0.9173	0.9152
3	3	50	0.9812	0.9894	0.9887
4	4	20	0.9610	0.9662	0.9985
5	5	13	0.9412	0.9504	0.9950

We used 500 epochs to train the model to classify flight delays. Figure 4-3 shows how the DNN model can correctly classify 99.82% of the delayed flights and 78.42% of the class that are not delayed. This classification contains more than 150,000 flight records, and the model correctly classified 128,609 of them in the delay class and 63,436 in the no delay classes, respectively, which is a very good performance for any flight delay decision support system.

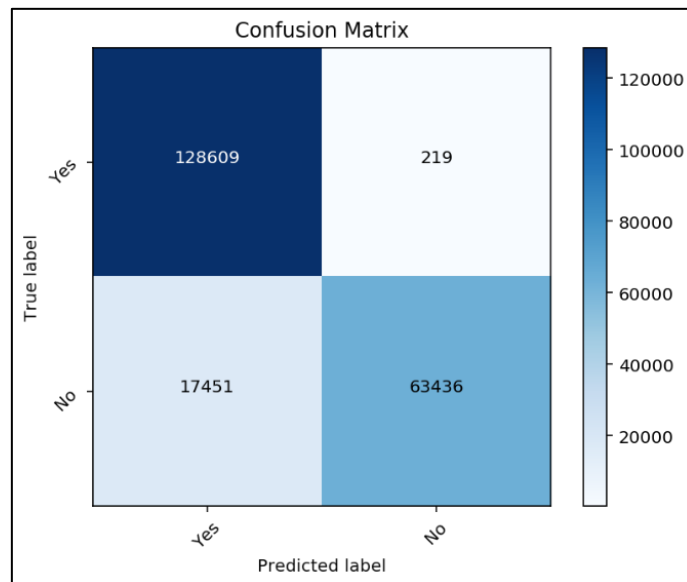


Figure 4-3 Confusion Matrix of Testing Classification using DNN with an Epoch=500

We then reduced the number of epochs to 200 when training the model for the same classification of flight delays. Figure 4-4 shows how the DNN model can correctly classify 99.45% of delayed flights and 81.54% of the class that are not delayed. This classification contains more than 150,000 flight records, and the model correctly classified 128,129 of them in the delay class and 65,958 in the no-delay class, respectively, which is again a very good performance for any flight delay decision support system.

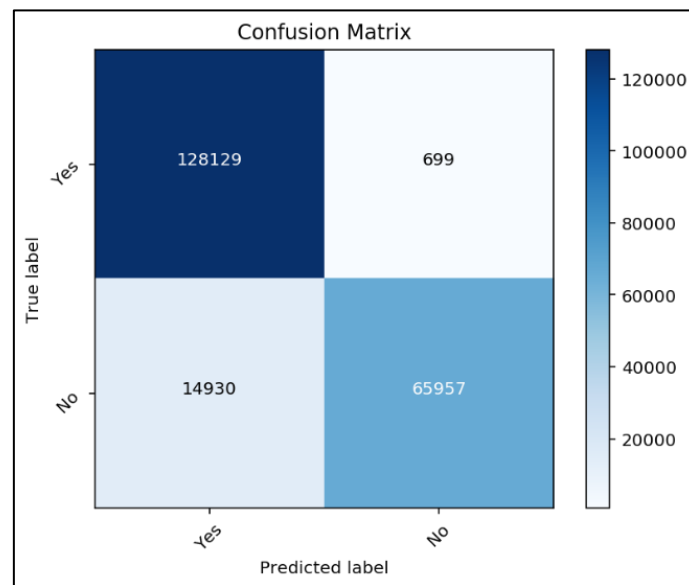


Figure 4-4 Confusion Matrix of Testing Classification using DNN with an Epoch=200

Figure 4-5 shows the training and validation accuracy of the DNN model. We observed that the model performs very well during the training and validation phases, with consistently good performance across every example of the training set, which is an advantage when developing a flight delay decision support system.

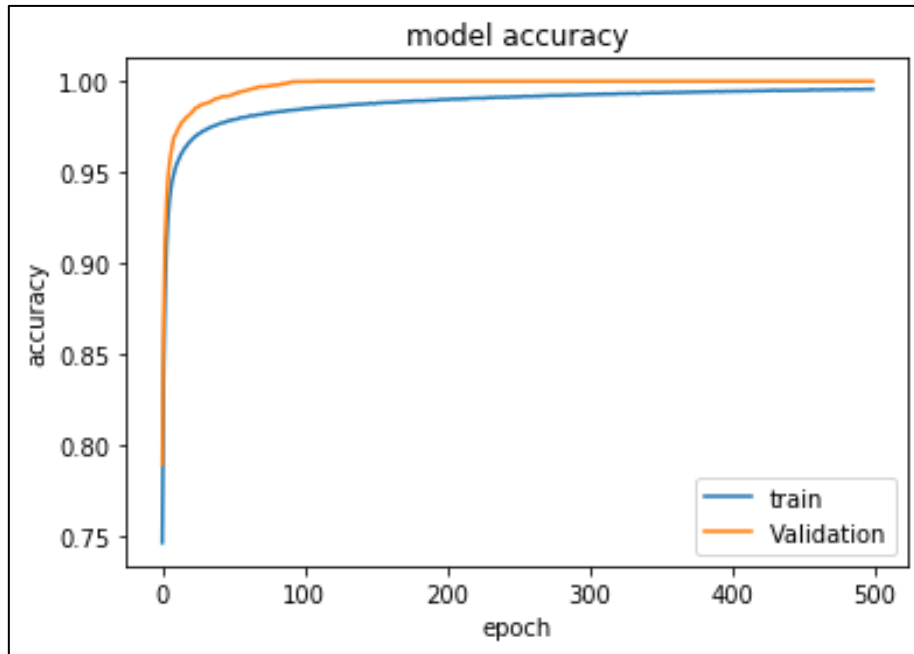


Figure 4-5 Model Accuracy Plot for DNN with Epoch=500

Figure 4-6 shows the training and validation loss of the DNN model. We observed that the model again performed very during the training and validation phases, with consistently good performance across every example of the training set, which is an advantage when developing a flight delay decision support system.

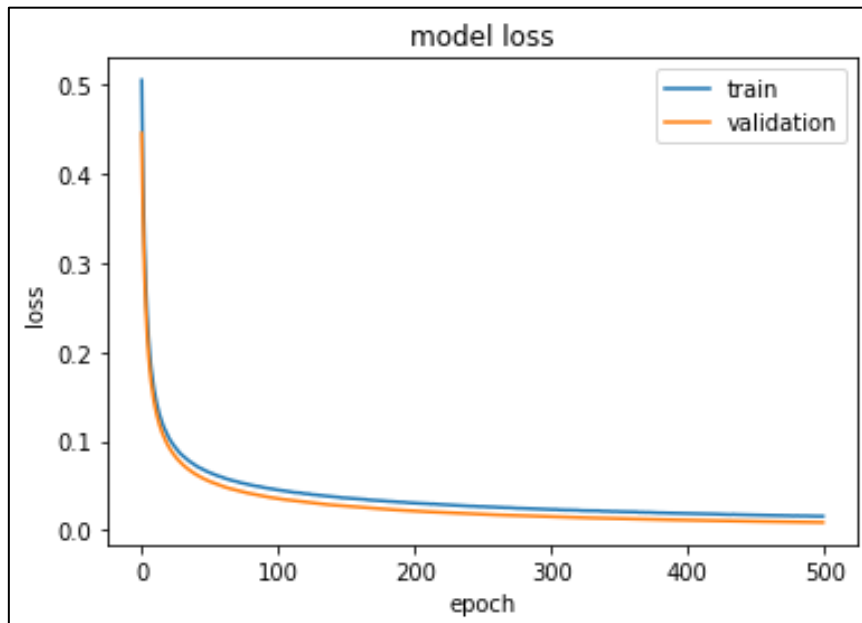


Figure 4-6 Model Loss Plot for DNN with Epoch=500

Figure 4-7 shows the final model of the DNN and the architecture with all the elements as applied in training the model with 500 epochs.

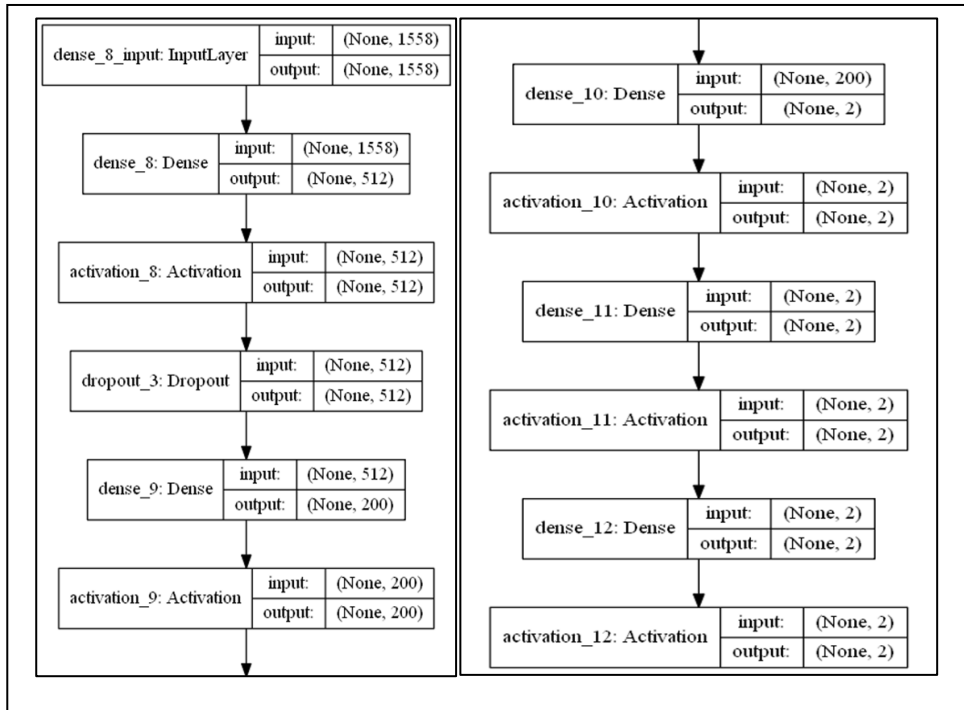


Figure 4-7 Final DNN Model Architecture

4.5 Conclusion and Future Direction

Major advances in Deep Learning have been recorded in solving problems that might have been difficult for years. Discovering the intricate structures in high-dimensional datasets made it possible. DNNs have recorded more successes and are predicted to have more robustness in the future of data because they largely require little engineering by hand. DNN largely take advantage of data and computational ability consequently. Despite this huge success, few have applied DNN in flight delay binary classification problems. Our research is among the few that have applied DNN and compared it with other shallow architectures, such as SVM and NN, to study their performance in a classification-related problem. We set out to clearly understand if the performance will keep track in other experiments by comparing the predictive accuracy of the widely used DNN with other shallow methods at different epochs

and parameter tunings. Our research aimed to identify the best conditions under which DNN performs better than SVM and NN.

Our experimental results show that the data size has a relative effect on the performance ability of the three classifiers, DNN, NN and SVM, at different parameter adjustments. However, the classification accuracy of DNN is better when it learns the underlying structure of the datasets. We observed that when the epochs were increased, there was a significant increase in predictive accuracy. The comparison of the performance of DNN, NN and SVM on classifying delays using the BTS on-time dataset is a topic of continuous research due to the need for more flexible flight delay predictions.

This paper has contributed to the aviation industry and the air transportation unit, specifically improving passenger experience through better flight delay decision support systems. Empirically, it is evident that most Deep Learning approaches are more likely to outperform the shallow architectures in a classification task due to their flexibility in parameter tuning. Our results show that DNN can be further investigated in other predictive analyses, such as aircraft maintenance. Finally, as evidenced by our experiments, we cannot claim that any algorithm is inferior to another on its merit. Instead, the ability of researchers to understand and tune these algorithms will go a long way in impacting their performance.

Acknowledgement & Funding

The authors wish to thank the reviewers for their constructive feedback. The Petroleum Trust Development Fund (PTDF) Nigeria partially funded this work through grant number PTDF/ED/OSS/PHD/DBB/1558/19 for the first author's PhD studies.

4.6 References

1. Tan YE., Teong KS., Shabbir M., Foo LK., Chua SL. Modelling flight delays in the presence of class imbalance. ACM International Conference Proceeding Series. 2018;; 186–191. Available at: DOI:10.1145/3299819.3299847
2. European Union Aviation Safety Agency. Artificial Intelligence Roadmap: A Human-Centric Approach to AI in Aviation. 2020; (February): 1–33. Available at: <https://www.easa.europa.eu/newsroom-and-events/news/easa-artificial-intelligence-roadmap-10-published>
3. Zhu X., Li L. Improved Flight Time Predictions for Fuel Loading Decisions of Scheduled Flights with a Deep Learning Approach. 2020; (11215119): 1–12. Available at: <https://arxiv.org/abs/2005.05684>
4. Akpinar MT., Karabacak ME. Data mining applications in civil aviation sector: State-of-art review. CEUR Workshop Proceedings. 2017; 1852: 18–25. Available at: DOI:10.1016/j.apergo.2015.07.010
5. Boggavarapu R., Agarwal P., Rohith Kumar DH. Aviation Delay Estimation using Deep Learning. 2019 4th International Conference on Information Systems and Computer Networks, ISCON 2019. IEEE; 2019; : 689–693. Available at: DOI:10.1109/ISCON47742.2019.9036276
6. Borse Y., Jain D., Sharma S., Vora V., Zaveri A., Professor A., et al. Flight Delay Prediction System. International Journal of Engineering Research and Technology (IJERT). 2020; 9(3): 88–92. Available at: www.ijert.org
7. Cheng F., Baszczewski B., Gulding J. A hybrid optimisation-simulation approach for itinerary generation. Proceedings - Winter Simulation Conference. IEEE; 2015. pp. 1885–1896. Available at: DOI:10.1109/WSC.2014.7020036
8. Airports Commission. Economy: Delay Impacts Assessment. 2014; (November).
9. Gui G., Liu F., Sun J., Yang J., Zhou Z., Zhao D. Flight delay prediction based on aviation data and Machine Learning. IEEE Transactions on Vehicular Technology. 2020; 69(1): 140–150. Available at: DOI:10.1109/TVT.2019.2954094
10. Balakrishna P., Ganesan R., Sherry L. Application of reinforcement learning algorithms for predicting taxi-out times. Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009. 2009; : 255–261.
11. Xiangmin G., Li M. Departure capacity prediction for hub airport in thunderstorm based on data mining method. Proceedings of the 29th Chinese Control and Decision Conference, CCDC 2017. 2017; : 6004–6009. Available at: DOI:10.1109/CCDC.2017.7978246
12. Choi S., Kim YJ., Briceno S., Mavris D. Prediction of weather-induced airline delays based on Machine Learning algorithms. AIAA/IEEE Digital Avionics Systems Conference - Proceedings. 2016. Available at: DOI:10.1109/DASC.2016.7777956
13. Belcastro L., Marozzo F., Talia D., Trunfio P. Using scalable data mining for predicting flight delays. ACM Transactions on Intelligent Systems and Technology. 2016; 8(1): 1–20. Available at: DOI:10.1145/2888402
14. Klein A., Kavoussi S., Lee RS. Weather forecast accuracy: Study of impact

- on airport capacity and estimation of avoidable costs. Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009. 2009. pp. 12–21.
15. Evans AD., Sridhar B., McNally D. Improving operational acceptability of dynamic weather routes through analysis of commonly used routings. 16th AIAA Aviation Technology, Integration, and Operations Conference. 2016. pp. 1–18. Available at: DOI:10.2514/6.2016-3600
 16. Thiagarajan B., Srinivasan L., Sharma AV., Sreekanthan D., Vijayaraghavan V. A Machine Learning approach for prediction of on-time performance of flights. AIAA/IEEE Digital Avionics Systems Conference - Proceedings. 2017; 2017-Sept: 1–6. Available at: DOI:10.1109/DASC.2017.8102138
 17. Lu L., Jin P., Pang G., Zhang Z., Karniadakis GE. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. 2021; 3(March): 31–33. Available at: DOI:10.1038/s42256-021-00302-5
 18. Lecun Y., Bengio Y., Hinton G. Deep Learning. Nature. 2015; 521(7553): 436–444. Available at: DOI:10.1038/nature14539
 19. Ciruelos C., Arranz A., Etxebarria I., Peces S., Campanelli B., Fleurquin P., et al. Modelling delay propagation trees for scheduled flights. Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2015. 2015. pp. 1–10. Available at: https://www.researchgate.net/profile/Victor-Eguiluz/publication/336837610_Modelling_Delay_Propagation_Trees_for_Scheduled_Flights/links/5db5bbe792851c577eceb30/Modelling-Delay-Propagation-Trees-for-Scheduled-Flights.pdf
 20. Bus F. The prediction of flight delays based the analysis of Random flight points. 2015; 00: 3992–3997.
 21. Kim YJ., Choi S., Briceno S., Mavris D. A Deep Learning approach to flight delay prediction. AIAA/IEEE Digital Avionics Systems Conference - Proceedings. IEEE; 2016. pp. 1–6. Available at: DOI:10.1109/DASC.2016.7778092
 22. Cai KQ., Zhang J., Xiao MM., Tang K., Du WB. Simultaneous optimisation of airspace congestion and flight delay in air traffic network flow management. IEEE Transactions on Intelligent Transportation Systems. 2017; 18(11): 3072–3082. Available at: DOI:10.1109/TITS.2017.2673247
 23. Proenca HM., Klijn R., Bäck T., Van Leeuwen M. Identifying flight delay patterns using diverse subgroup discovery. Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence, SSCI 2018. 2019. pp. 60–67. Available at: DOI:10.1109/SSCI.2018.8628933
 24. Manna S., Biswas S., Kundu R., Rakshit S., Gupta P., Barman S. A statistical approach to predict flight delay using gradient boosted decision tree. ICCIDS 2017 - International Conference on Computational Intelligence in Data Science, Proceedings. 2018; 2018-Janua: 1–5. Available at: DOI:10.1109/ICCIDS.2017.8272656
 25. Venkatesh V., Arya A., Agarwal P., Lakshmi S., Balana S. Iterative machine and Deep Learning approach for aviation delay prediction. 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer

- and Electronics, UPCON 2017. 2017. Available at: DOI:10.1109/UPCON.2017.8251111
26. Levi Y., Bekhor S., Rosenfeld Y. A multi-objective optimisation model for urban planning: The case of a very large floating structure. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November 2018): 85–100. Available at: DOI:10.1016/j.trc.2018.11.013
 27. Zeng N., Qiu H., Wang Z., Liu W., Zhang H., Li Y. A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of Alzheimer's disease. *Neurocomputing*. Elsevier B.V.; 2018; 320: 195–202. Available at: DOI:10.1016/j.neucom.2018.09.001
 28. Wu W., Cai K., Yan Y., Li Y. An Improved SVM Model for Flight Delay Prediction. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. 2019; 2019-Septe. Available at: DOI:10.1109/DASC43569.2019.9081611
 29. Lukas S., Stefani D., Widjaja P. Comparing SVM and GLM in calculating insurance premium for flight delay. *ACM International Conference Proceeding Series*. 2019. pp. 141–145. Available at: DOI:10.1145/3369114.3369160
 30. Xu Y., Liu L., Gao X., Zeng FF. Analysis of Factors in Flight Delay. 2019; 93(Mmsta): 170–176. Available at: DOI:10.2991/mmsta-19.2019.36
 31. Moreano JC., Palomino NBL. Global facial recognition using gabor wavelet, support vector machines and 3d face models. *Journal of Advances in Information Technology*. 2020; 11(3): 143–148. Available at: DOI:10.12720/jait.11.3.143-148
 32. Ben Taieb S., Hyndman RJ. Recursive and direct multi-step forecasting: the best of both worlds. *International Journal of Forecasting*. 2014; (September).
 33. Kadhum M., Manaseer S., Dalhoum ALA. Evaluation feature selection technique on classification by using evolutionary ELM wrapper method with features priorities. *Journal of Advances in Information Technology*. 2021; 12(1): 21–28. Available at: DOI:10.12720/jait.12.1.21-28
 34. Neuron A., Hammer B., Iliadis L., Hutchison D., Caterini AL., Chang DE., et al. and Machine Learning – ICANN 2018. Evidence Based Midwifery. 2018. Available at: DOI:10.1023/A
 35. Orimoloye LO., Sung MC., Ma T., Johnson JEV. Comparing the effectiveness of deep feedforward neural networks and shallow architectures for predicting stock price indices. *Expert Systems with Applications*. Elsevier Ltd; 2020; 139: 112828. Available at: DOI:10.1016/j.eswa.2019.112828
 36. Nguyen NN., Duong AT. Comparison of two main approaches for handling imbalanced data in churn prediction problem. *Journal of Advances in Information Technology*. 2020; 12(1): 29–35. Available at: DOI:10.12720/jait.12.1.29-35
 37. Schmidhuber J. Deep Learning in neural networks: An overview. *Neural Networks*. 2015; 61: 85–117. Available at: DOI:10.1016/j.neunet.2014.09.003
 38. van Cranenburgh S., Alwosheel A. An artificial neural network based approach to investigate travellers' decision rules. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November 2018): 152–

166. Available at: DOI:10.1016/j.trc.2018.11.014
39. Orsini F., Gastaldi M., Mantecchini L., Rossi R. Neural networks trained with WiFi traces to predict airport passenger behavior. MT-ITS 2019 - 6th International Conference on Models and Technologies for Intelligent Transportation Systems. IEEE; 2019; : 1–7. Available at: DOI:10.1109/MTITS.2019.8883365
40. Koprinkova-Hristova P., Mladenov V., Kasabov NK. Artificial Neural Networks - Methods and Applications in Bio-/Neuroinformatics: Preface. Artificial Neural Networks - Methods and Applications in Bio-/Neuroinformatics. 2015. V p.
41. Bunrit S., Kerdprasop N., Kerdprasop K. Improving the representation of cnn based features by autoencoder for a task of construction material image classification. Journal of Advances in Information Technology. 2020; 11(4): 192–199. Available at: DOI:10.12720/jait.11.4.192-199
42. Krizhevsky BA., Sutskever I., Hinton GE. Imagenet classification with deep convolutional neural networks. Imagenet classification with deep convolutional neural networks. Communications of the ACM. 2012; 60(6): 84–90.
43. An Introduction to Deep Feedforward Neural Networks _ by Reza Bagheri _ Towards Data Science.
44. Zintgraf LM., Cohen TS., Adel T., Welling M., Brussel VU. Flight delay error analysis for amsterdam airport schiphol using a deep neural network (dnn). 2017; (January): 1–12.
45. Hu C., Wei H., B MC. Enhancing Airlines Delay Prediction by Implementing Classification Based Deep Learning Algorithms. Springer International Publishing; 2019. 563–581 p. Available at: DOI:10.1007/978-3-030-19063-7.
46. Alawad H., Kaewunruen S. Wireless sensor networks: Toward smarter railway stations. Infrastructures. 2018; 3(3): 1–17. Available at: DOI:10.3390/infrastructures3030024
47. Dindar S., Kaewunruen S., An M., Sussman JM. Bayesian Network-based probability analysis of train derailments caused by various extreme weather patterns on railway turnouts. Safety Science. 2018. pp. 20–30. Available at: DOI:10.1016/j.ssci.2017.12.028
48. Ngamkhanong C., Kaewunruen S. Prediction of Thermal-Induced Buckling Failures of Ballasted Railway Tracks Using Artificial Neural Network (ANN). International Journal of Structural Stability and Dynamics. 2022. Available at: DOI:10.1142/S0219455422500493
49. Alawad H., Kaewunruen S., An M. A Deep Learning Approach towards Railway Safety Risk Assessment. IEEE Access. 2020; 8: 102811–102832. Available at: DOI:10.1109/ACCESS.2020.2997946
50. Kaewunruen S., Sresakoolchai J., Huang J., Zhu Y., Ngamkhanong C., Remennikov AM. Machine Learning Based Design of Railway Prestressed Concrete Sleepers. Applied Sciences (Switzerland). 2022; 12(20). Available at: DOI:10.3390/app122010311
51. Sresakoolchai J., Hamarat M., Kaewunruen S. Prediction of turnout support deterioration through dynamic train-track interactions integrated with Artificial Intelligence. Internoise 2022 - 51st International Congress and

- Exposition on Noise Control Engineering. Institute of Noise Control Engineering; 2022. pp. 5375–5382. Available at: DOI:10.3397/in_2022_0782
52. Liang R., Liu W., Kaewunruen S., Zhang H., Wu Z. Classification of External Vibration Sources through Data-Driven Models Using Hybrid CNNs and LSTMs. *Structural Control and Health Monitoring*. 2023; 2023(ii): 1–18. Available at: DOI:10.1155/2023/1900447
 53. Lavercombe A., Huang X., Kaewunruen S. Machine Learning application to eco-friendly concrete design for decarbonisation. *Sustainability (Switzerland)*. 2021; 13(24). Available at: DOI:10.3390/su132413663
 54. Kaewunruen S., Sresakoolchai J., Stittle H. Machine Learning to Identify Dynamic Properties of Railway Track Components. *International Journal of Structural Stability and Dynamics*. 2022. Available at: DOI:10.1142/S0219455422501097
 55. Kaewunruen S., Sresakoolchai J., Zhu G. Machine Learning Aided Rail Corrugation Monitoring for Railway Track Maintenance. *Structural Monitoring and Maintenance*. 2021; 8(2): 151–166. Available at: DOI:10.12989/smm.2021.8.2.151
 56. Rungskunroch P., Shen ZJ., Kaewunruen S. Benchmarking Socio-Economic Impacts of High-Speed Rail Networks Using K-Nearest Neighbour and Pearson's Correlation Coefficient Techniques through Computational Model-Based Analysis. *Applied Sciences (Switzerland)*. 2022; 12(3). Available at: DOI:10.3390/app12031520
 57. Huang X., Zhang J., Sresakoolchai J., Kaewunruen S. Machine Learning aided design and prediction of environmentally friendly rubberised concrete. *Sustainability (Switzerland)*. 2021; 13(4): 1–27. Available at: DOI:10.3390/su13041691

Chapter 5: An Enhanced Deep Autoencoder for Flight Delay Prediction

Accurate and timely flight delay prediction cannot be overemphasised because of the ever-increasing demand for air travel and its importance in deploying intelligent transportation systems. Nonetheless, there has not been a universal solution to the problem, as more intelligent flight decision systems are required for the aviation industry's future growth. Existing flight delay classification and prediction approaches are mainly shallow traffic models and do not satisfy many applications in the real world. Our motivation to rethink the deep architecture model for predicting flight delays emanates from the problem. In this research, we proposed a technique that modified stacked autoencoder architecture parameters for training the network and understanding the link between space, time and information gained from the flight on-time data. We developed three different types of autoencoders based on the architecture of the modified stacked autoencoder. The models learn the generic flight delay features, and it's trained greedily in a layer-wise fashion. This paper is among the first to compare the performances of vanilla autoencoder, logistic regression autoencoder and multilayer perceptron autoencoder based on the developed modified stacked autoencoder architecture. Moreover, our experiment demonstrates that the models achieved varying levels of accuracy in the flight delay classifications task. The deep vanilla autoencoder shows superior accuracy, recall and precision performance compared to logistic regression autoencoder and multilayer perceptron autoencoder at different parameter settings.

5.1 Introduction

Recently, social life has become heavily reliant on the aviation industry. Precise flight delay categorisation is profitable but challenging due to the uncertainties and breakdowns inside the procedures. Even though people can travel by car or boat, air travel is superior for most passengers due to its efficiency and reliability [1]. Airlines, airports and passengers lose when flights are delayed. All actors in the commercial aviation industry rely on their predictions when making decisions.

Furthermore, given the complexity of the air transport industry, the number of approaches for prediction, and the stream of flight data, developing accurate prediction models for flight delays has become difficult. The transit delay is one of the most frequently recalled performance measurements. Commercial aviation players

define delay as the time it takes for a flight to arrive/depart late or rescheduled. The discrepancy between the schedule and actual flight arrival or departure times indicates the delay. Flight delay is, without a doubt, an important topic from the perspective of air transport networks. In 2013, 36% of flights in Europe were delayed by more than 5 minutes, 31.1% of flights in the United States were late by more than 15 minutes, and 16.3% of flights in Brazil were cancelled or had delays of more than 30 minutes [2]. Flight delays can be caused by many factors, such as extreme weather (0.72%), late-arriving aircraft (7.86%), air carriers (5.90%), National Aviation System (7.19%), and security (0.04%), as reported in 2017 by the Bureau of Transportation Statistics (BTS) on filing on-time performance of carriers in the USA [3–6].

Passengers, airlines, and airports suffer from flight delays, which greatly impact the economy. Passengers frequently plan to arrive several hours early for their appointments, raising travel expenditures because of the unpredictability of flight delay occurrence [7]. Airlines face fines, sanctions, and increased operating costs, such as airport personnel and aircraft holding [8]. Delays may also harm the environment by raising fuel usage and gas emissions, which is a concern from a sustainability standpoint [9]. Delays can threaten airline marketing initiatives, as carriers rely on customers' loyalty to maintain frequent flyer programmes, and consistent performance influences consumer choice. There is a link between the number of delays and prices, the aircraft size, the regularity of flights, and customer complaints about airline service [10]. Estimating flight delays can help airport and airline officials make better management and strategic judgments and alert customers so they can reschedule their arrangements [11]. To better understand the overall flight environment, massive amounts of data from commercial aviation are always collected and kept in databases. Data scientists and analysts are improving their computational and data management abilities to extract usable information from the vast amounts of data produced by sensors and IoT [12, 13, 14].

This research contributes to the commercial air transport industry by examining flight delay prediction from a Machine Learning and data science viewpoint. It synthesises the most widely studied trends in predicting flight delays, especially the non-weather impacted flight delays, describes how this problem is tackled and compares strategies for developing effective flight delay prediction models. Despite several attempts by researchers and industries to find lasting solutions to flight

delays, there is no lasting solution to the problem, as the aviation industry's future expansion will demand more intelligent flight decision systems. In this paper, we proposed an approach for flight delay prediction for departure and arrival flights using the US flight on-time dataset. Our technique uses a deep autoencoder with re-scale parameters to predict non-weather-impacted flight delays. We develop and investigate the performance of three approaches: multilayer perceptron autoencoder, vanilla autoencoder and logistic regression. The approach uses a deep-stacked autoencoder architecture. We re-scaled the hidden layers of the autoencoder architecture. We then used the architecture in training the models to check the performance of the approaches using accuracy, precision and recall metrics, respectively.

The rest of the paper follows: Section 5.2 presents the related work; Section 5.3 discusses the materials and method. Section 5.4 presents the results and discussion, and Section 5.5 has the conclusion and future direction.

5.2 Related Work

For several decades, accurate flight delay prediction has become challenging for practitioners and researchers. However, the advent of high computational methods and algorithms has improved the performance of Deep Learning methods demonstrating state-of-the-art results compared to statistical or classical Machine Learning techniques evaluated on their prediction accuracy, precision and recall.

The authors in [15] proposed a Deep Learning-based prediction approach that uses a stacked autoencoder model to predict the time of an airport delay. The research solidifies Deep Learning theory and implementation in aviation and supports civil aviation transportation management control. When choosing variables, they consider the complex relationship of airline delay levels in time and space and implicitly define traffic flow's network character and spatial-temporal correlations. The data used in the study comes from the U.S. Federal Aviation Administration (FAA) official website database. It includes the date, time, runway location, departure and arrival delay rate, departure and arrival delay flight, and GMT hours for American airports from January 2016 to December 2017. The authors in [16] proposed a new Deep Learning-based optimised forecasting model that uses the Levenberg-Marquart (LM) algorithm to predict flight delays. The study's main goal was to reduce the number of data dimensions before loading them into a Deep Learning network. The positive effect of

denoising autoencoder and LM algorithm was validated by creating two other structures. The first removed the denoising autoencoder, while the latter removed the LM algorithm. Furthermore, to balance the data, they used under-sampling and over-sampling techniques. They concluded that optimising the results with a stack denoising autoencoder and the LM algorithm and balancing the dataset positively impacts delay forecasting and increases the accuracy and precision of the denoising autoencoder model.

The authors in [17] created a two-stage predictive model using supervised Machine Learning techniques to forecast on-time flight performance. The model's first stage uses binary classification to forecast flight delays, while the second stage uses regression to forecast the duration of the delay in minutes. Gradient Boosting Classifier outperformed Extra-Trees Regressor in the classification stage, while in the regression stage, it was the opposite. In [18], Airports are treated as nodes, while direct flights serve as linkages between them, with delays manifesting as errors in the daily schedule that can spread across a significant portion of the network. The delay propagation process is simulated using an agent-based, data-driven methodology. The initial results resemble genuine delay propagation patterns, allowing the cluster of crowded airports to be described and its progression during the day. The authors [19] provide a series analysis-based influence model approach of arbitrary flight points, integrated with a Bayesian Network and a Gaussian mixture model-expectation maximisation algorithm. As a result, they can estimate the downstream flight delay conditions in real-time when the upstream flight is delayed. A series of experiments reveals high accuracy, and the solution has good analytical and prediction dependability.

In [20], the authors look at the usefulness of Deep Learning models in predicting air traffic delays. The initial step is to use deep RNN to estimate the daily delay status. The next step is to anticipate delayed flights using the daily delay status generated during the first step, historical on-time performance data, and weather data. An accurate delay status of a single day may be obtained by adding deep Long Short-Term Memory RNN architecture to the forecasting model. Flights' most precise delay states were obtained by entering a day's delay status into an individual flight delay model. The multi-objective optimisation problem is described in [21] as follows: It aims for the best possible flight routes as well as arrivals and departures timeslots for all scheduled flights, intending to reduce airspace overcrowding and total delayed flights,

given an air traffic network, time frame, the correlating airspace sector structure and initial flight plans between various airports. To identify flight delay characteristics from data, in [22], the authors propose using subgroup detection, a data mining methodology that allows the discovery of subgroups or subsets of data that somehow differ from the total data. A conjunctive query on the remaining variables can be used to explain it. They investigate aircraft delays using characteristics that are accessible in advance and also employ subgroup identification to find regions of interest in the dataset. The findings imply that irrelevant criteria like the day of the week and month can identify huge subsets of flights with relatively long delays.

The effectiveness of this established model for flight delay prediction tasks is investigated in [23]. A precise and robust prediction model has been constructed by merging this regression model built on the Machine Learning approach, allowing for an in-depth investigation of air traffic delay patterns. For predicting flight delays, the Gradient Boosted Decision Tree model was used. The model has the greatest Coefficient of Determination of 92.31% and 94.85% for arrival and departure, respectively, for the training dataset. In [24], a Machine Learning-based methodology is proposed for predicting airline arrival delays by considering input characteristics varying from distance to weather conditions to determine whether a certain flight is delayed. To estimate flight delay, it employs neural networks and Deep Learning techniques. This paper is key to the effectiveness of employing neural networks and deep architectures to classify flight delay or no delay. The results show that the prediction accuracy employing neural networks is 92%, with one hidden layer comprising three neurons, while with four neurons for each of the two hidden layers, Deep Networks achieved an accuracy of 77%. In [25], the major goal of the study was to determine the ideal conditions in which Deep Feedforward Neural Network outperforms a Support Vector Machine and Neural Network. By comparing the predicted efficiency of the extensively used Deep Neural Network with other superficial approaches at various epochs and variable tunings, they set out to fully comprehend if the performance will maintain track in further studies. The authors discovered that increasing the epochs resulted in a considerable boost in predicting accuracy.

The goal in [26] is to develop cost-sensitive classifiers that can forecast flight delays. Authors in [27] proposed a classification method that creates a single rule for every prediction in the dataset and then chooses the rule with the fewest overall error as its only rule. Data analysis will reveal that delayed flights have distinct patterns from

on-time flights. Their technology also provides weather forecast information and the likelihood of a weather delay. The major goal in [28] is to forecast the departure delay on a specific link or at a particular airport in the future. The Random Forest algorithms predicted future departure delays of up to 24 hours. The algorithm is a multi-decision tree ensemble classifier that reports the mode of the classifications generated by individual trees. According to the analysis, connection delays' reliance on network status ranged from link to link. The classification and regression approaches were quite resilient to changes in the forecast horizon. Using the RL algorithm in [29] to anticipate taxi-out timings at airports using multi-dependent static surface surveillance data will increase prediction accuracy. The RL estimator predicts taxi-out time based on the dynamic system state. A Markov decision process represents the choice to anticipate the taxi-out time based on the system status. The prediction approach outlined in this study allows one to forecast average airport taxi-out predictive factors 30-60 minutes ahead of time for any given moment. In [3], the authors used a nonparametric RL-based technique placed in the probabilistic framework of stochastic dynamic programming to evaluate the accuracy of taxi-out time prediction. The findings show that the RL estimator can reasonably capture the dynamics at even the most difficult airports.

In [30], the research presents a new Machine Learning-based strategy for detecting large-scale aircraft delays. This proposed method begins with unsupervised learning on flight data gathered from the airport. After the learning process, the standard for each delay category can be determined. The supervised learning method can build the alert model using these delays. The authors developed a new form of multilevel input layer Artificial Neural Network (ANN) in [31] that can manage nominal variables and is open to interpretation in the view that the links among distinct input and output variables can be seen. They compared the suggested approach to the standard gradient descent backpropagation ANN model and found that the introduced model outperformed the classic backpropagation method regarding prediction error and training time. In [32], the authors evaluate the results of various approaches to forecasting delays in air traffic networks. They use standard network architectures, classification, regression trees, and standard linear regression. The ANN scored well at Origin-Destination pair delay classification, but the Markov Jump Linear System is best for the delay and airport delay regression. Time of day became irrelevant at

extended forecast ranges for the Origin-Departure pair delay regression problem, whereas network characteristics were essential.

Because of the remarkable performance recorded recently by Deep Learning methods in a predictive task, most especially in the area of delay predictive tasks, this paper studies the prediction of delay and proposes a procedure by re-scaling autoencoder parameter through a nested multiplication of the hidden layers with a single drop-out on the final hidden layer before feeding the output layer and then compare it on three different types of autoencoders. All the revised studies employed and proposed models handling flight delays with a concentration on improving the performance through data pre-processing with Machine Learning or classical deep autoencoders. However, no study has applied a re-scaling parameter that compares the performance of three models developed in this research by modifying them on a stacked autoencoder architecture. Therefore, this paper aims to investigate the efficacy and viability of a re-scaled autoencoder parameter on three different types of autoencoders to model and predict flights' on-time performance. This research intends to provide valuable insight for accurately predicting flight delays in the air transport system and the aviation industry.

5.3 Materials and Methods

There are many interesting applications of autoencoders in different fields, such as visualisation, dimensionality reduction and data compression. Recently, It has been discovered to have the capability of training neural networks because of its difference in the gradients magnitudes of the lower and higher layers and the lack of generalisation of deep networks due to many parameters in the deep neural network. Also, the objective function curvature is difficult to find a good local minimum. Pretraining the network helps divide the entire deep network into steps. Pretraining the deep network is intended to address the difficulties mentioned above, and the training process of the deep network is divided into steps such as the pretraining step, where shallow autoencoders sequence use unsupervised data one layer greedily at a time. Training the last layer with a supervised dataset and finally fine-tuning the entire network with supervised data and backpropagation.

The autoencoder can learn data features and represent the statistical characteristics of training samples. The input of the model is its target output [13]. The autoencoder contains an encoder and a decoder. The encoder shrinks the input while

the decoder tries to reconstruct the input from the encoder's compressed form. The encoder model is saved after training, whereas the decoder is deleted. The encoder can then be used as a data preparation approach to extract features from raw data so that a new Machine Learning model [33, 34, 43, 35–42]. An autoencoder with a hidden layer and an output layer is shown in Figure 5-1 [13].

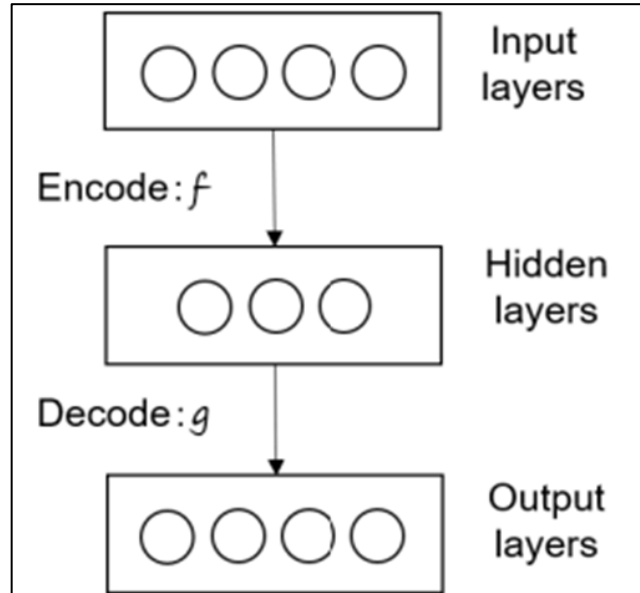


Figure 5-1 Structure of Autoencoders

The reconstruction process of an input x of an autoencoder can be explained using Equation (5-1), Equation (5-2), Equation (5-3), Equation (5-4) and Equation (5-5), respectively.

Assume we have a set of data $x_{(i)}$ belonging to the set of real numbers at points $\{x_{(1)}, x_{(2)}, \dots, x_{(m)}\}$ with many dimensions. Mapping the set of the input to another data point set $z_{(i)}$ with points $\{z_{(1)}, z_{(2)}, \dots, z_{(m)}\}$ having a lower dimension than $x_{(i)}$ and can be reconstructed back to $x_{(i)}$ effectively. Mapping the data back and forth in a systematic manner can be achieved by encoding the input layer $x_{(i)}$ into a hidden layer $z(x_{(i)})$ and $\bar{x}(z_{(i)})$ is the reconstructed result shown in Equations (5-1) and (5-2), respectively.

$$z(x_{(i)}) = w_1 x_{(i)} + c_1 \quad (5-1)$$

$$\bar{x}(z_{(i)}) = w_2 z_{(i)} + c_2 \quad (5-2)$$

Where W_1 and C_1 are the weighted matrices and coding bias vectors, and W_2 and C_2 are the decoding and coding bias matrices, respectively. To minimise stochastic

gradient descent, the reconstruction error of approximating $x_{(i)}$ by $\bar{x}_{(i)}$. We set an objective function which sums the squared of the differences between $x_{(i)}$ and $\bar{x}_{(i)}$:

$$L(W_1, C_1, W_2, C_2) = \sum_{i=1}^n (\bar{x}_{(i)} - x_{(i)})^2 \quad (5-3)$$

$$= \sum_{i=1}^n (W_2 z_{(i)} + C_2 - x_{(i)})^2 \quad (5-4)$$

$$= \sum_{i=1}^n W_2 (W_1 x_{(i)} + C_1) + C_2 - x_{(i)}^2 \quad (5-5)$$

We are looking closely at how an autoencoder works. It takes a high-dimensional input, such as a picture or a vector, and runs it through a neural network that compresses the data into a shorter form with two principal components. The first is the encoder, a collection of fully connected or convolutional layers that compress the input into a smaller representation with fewer dimensions than the data, a process known as a bottleneck. It reconstructs the input using this bottleneck's fully connected or convolutional layers.

We employed the sigmoid activation function in the deep neural network. It is a non-linear AF that outputs a continuous value between 0 and 1, defined as shown in Equation (5-6):

$$a = g(z) \frac{1}{1 + e^{-z}} \quad (5-6)$$

The sigmoid function is like the step function in that it is continuous and avoids the leap in output values that the step function does [44–47]. Figure 5-2 shows a plot of the sigmoid AF [7, 37, 48–51].

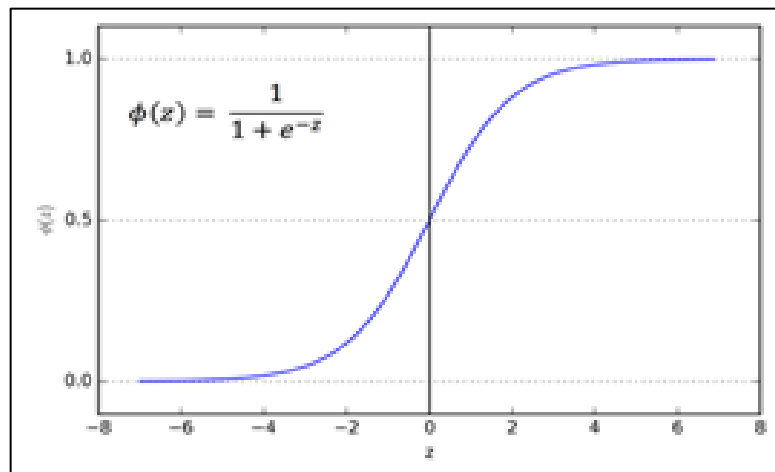


Figure 5-2 Sigmoid Activation Function

5.3.1 Proposed Architecture

Our proposed architecture is based on a stacked autoencoder design. We modified the stacked autoencoder, which is a deep neural network made up of many layers of sparse autoencoders, with each hidden layer's output coupled to the input of the next hidden layer, where we nested all hidden layers with a single drop-out and re-scaled by multiplying the current layer with the output of the previous layer before feeding the next layer for the purpose of normalisation. The last hidden layer is composed of a drop-out to address any overfitting or underfitting of the model.

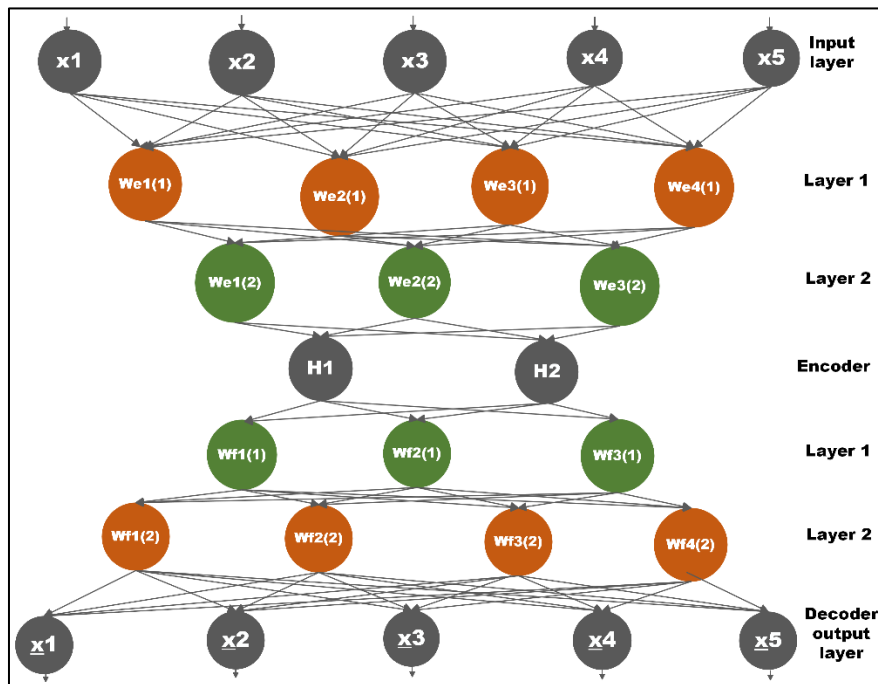


Figure 5-3 The Proposed Architecture

The hidden layers are trained using an unsupervised technique and subsequently fine-tuned using a supervised method, as shown in Figure 5-3. A regular predictor should be added to the top layer of the network in order to use the deep autoencoder network for flight delay prediction. The basic structure of a stacked autoencoder is as follows:

- Train the autoencoder and acquire the taught data using input data.
- The preceding layer's learned data is utilised as input for the next layer and works until the training is finished.
- Once all the hidden layers have been trained, the cost function is minimised using the backpropagation technique, and weights are adjusted with the training data for fine-tuning.

5.3.2 Experimentation

We developed our proposed stacked autoencoder network architecture using multilayer perceptron, vanilla and logistic regression.

Multilayer Perceptron Autoencoder

In this approach, rebuilding the model feature can be neglected after fitting the model, and the model can be used up to the bottleneck layer. The model produces a fixed-length vector with a compressed version of the input data at the bottleneck layer.

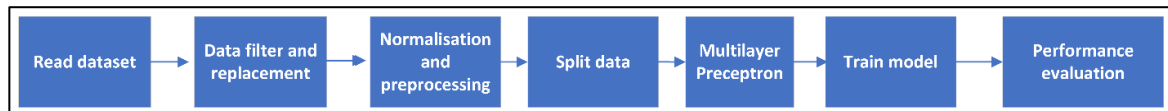


Figure 5-4 Multilayer Perceptron Autoencoder

Figure 5-4 shows the entire process of the multilayer perceptron autoencoder. It starts taking the input from the provided dataset and filtering it due to memory concerns. The data is divided into 70% train and 30% test sets before designing and fitting the model, and the input is scaled by normalising the values range between 0-1. The encoder is defined with various numbers of hidden layers. A similar structure will determine the decoder but in a reverse manner. Sigmoid activation and batch normalisation ensure that the model learns well. Since reconstruction is a multi-output regression issue, the model will be trained using the Adam version of stochastic gradient descent, which minimises the mean squared error. After training, it plots the loss for the train and test sets to confirm the model learned the classification problem, the precision-recall curve and the precision and recall for different threshold values.

Logistic Regression Autoencoder

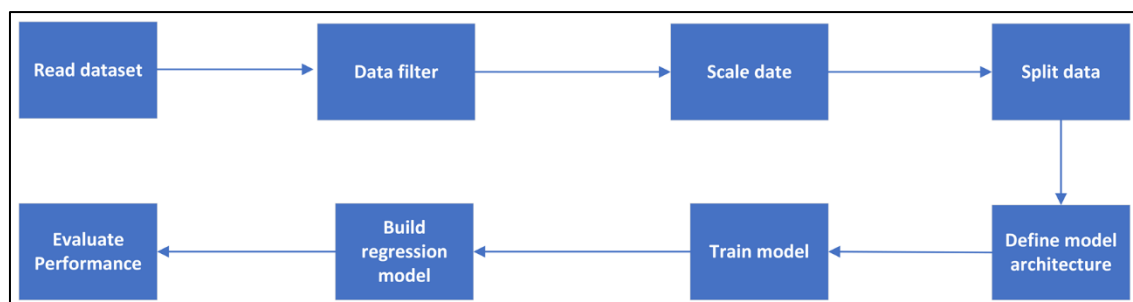


Figure 5-5 Logistic Regression Autoencoder

The approach takes the input from the dataset, as in Figure 5-5. The data is scaled and filtered to make it suitable for the autoencoder. The model consists of an input layer, with the size of the input variables taken from the dataset. The encoder

comprises several hidden layers to observe and analyse the model's behaviour when the number of hidden layers changes. The output layer takes the input from the last hidden layer and is retained to decode and visualise the encoded data. Data encoding can aid in creating a linear classification border for the data. The Logistic Regression model is used on the encoded data. It continuously creates training and testing data from the original and encoded data, makes the Logistic Regression model, and assesses its effectiveness. After it builds the Logistic Regression model, it plots the loss for the 70% train and 30% test sets to confirm the model learned the classification problem, the precision-recall curve and the precision and recall for different threshold values.

Vanilla Autoencoder



Figure 5-6 Vanilla Autoencoder

The input and output are the same in the vanilla autoencoder, the first being reconstructed using the Adam optimiser and the mean squared error loss function. As in Figure 5-6, it starts by reading the input from the dataset, filtering and replacing the input names with specific variables names used for the vanilla autoencoder, such as 1, 2,...n In the pre-processing stage, normalisation takes an important role in scaling down the range of data before it is used in the next stage. The data is again split into 70% train and 30% test sets, and the class label is dropped since it is an unsupervised method. As in the other two approaches, the autoencoder consists of an encoder and a decoder, each having multiple hidden layers with a specific hidden size. The decoder takes the information from the last hidden layer and reproduces the classification. After the model is trained, it plots the loss for the train and test sets to confirm that the model learned the classification problem, precision-recall curve, and recall for different threshold values. It ends with the confusion matrix, which shows how the results were classified.

5.3.3 Dataset Collection and Description

First and foremost, we selected model inputs at the start of the phase to learn and produce the final structure based on them. The dataset used to evaluate the model was compiled from historical data, including flight schedule information for three months, from October to December 2000. Table 5-1 lists the variables that are utilised

as inputs. There are 23 factors in the original dataset. However, certain elements were eliminated since they were irrelevant to the desired outcome. The data was provided by a real-world source from the United States (US) Bureau of Transportation Statistics.

Table 5-1 Dataset features

S/No.	Input	Reference	Type
1	Origin/Destination airport	3 letters format, e.g. LAX -Los Angeles International Airport	String
2	Planned Departure and Arrival Time	3/4 digits format, e.g. 1520 – 03:20 pm	Integer
3	Flight Number	4 digits format, 1451	Integer
4	Real Departure and Arrival Time	3/4 digits format, e.g. 1520 – 03:20 pm	Integer
5	Unique Carrier	2/3 letters format, e.g. PS	String
6	Elapse Time	2 digits format, e.g. 94	Integer
7	Departure/Arrival Delay	2 digits format, e.g. 11	Integer

Dataset Pre-processing

We cleaned the data by removing the “missing values” and replacing them with zeroes, while the inputs with less than 15 minutes of delay time and other irrelevant labels were dropped. The “Delayed” label was created to track late flights for arrival and departure.

Software Settings for Experiment

The implementation was done in Jupyter Notebook using Python programming language. It is a free web programme that lets users create and share scripts with live software, equations, visualisations, and texts. Data cleaning and transformation, statistical modelling, numerical simulation, data visualisation, Machine Learning, and other applications are just a few of the potential use of the tool [52, 53, 62, 54–61]. The relevant libraries used in the script can be found in Table 5-2.

Table 5-2 Libraries

S/No.	Library	Version
1	Keras	2.4.3
2	Numpy	1.19.2
3	Pandas	1.1.3
4	TensorFlow	2.5.0
5	Itertools	0.25.1
6	Matplotlib	3.3.2
7	Sklearn	0.23.2

5.3.4 Evaluation Metrics

Accuracy, precision and recall are the most important metrics for evaluating the performance of categorisation Machine Learning models. A variety of indicators were used to examine the effectiveness of the outcomes. Several notions have to be introduced, such as True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) in computing the model performance [34, 50, 63–66].

The simplest intuitive performance metric is accuracy, the ratio of accurately predicted observations to all observations, as shown in Equation (5-7).

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (5-7)$$

The ratio of accurately predicted positive observations to total expected positive observations is precision, as shown in Equation (5-8).

$$\text{Precision} = \frac{TP}{TP+FP} \quad (5-8)$$

The ratio of accurately predicted positive data to all findings in the actual class is known as recall, as shown in Equation (5-9).

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5-9)$$

Precision-Recall is a helpful metric of prediction success when the classes are severely unbalanced. Precision measures result from usefulness in data collection, while recall indicates how many properly relevant results are retrieved.

For various thresholds, the precision-recall curve depicts the trade-off between them. A large area below the curve indicates good recall and precision, with high precision indicating a low false-positive rate and high recall indicating a low false-negative rate. Both indicate that the classification produces precise results [33, 34, 36, 43, 67, 68]

5.4 Results and Discussion

The program is written in Python programming language and runs on local machines or Crescent HPC Cluster. On Crescent, it operates on GPU nodes with two Intel E5-2698 v3 CPUs with 32 CPU cores, totalling 256GB of shared memory. The experiments were repeated with varied input settings to see how the accuracy and loss changed and get the best prediction possible.

We evaluated the performance of multilayer perceptron autoencoder, logistic regression autoencoder and vanilla autoencoder in classifying and predicting flight delays to determine which strategy performs best in these situations. A precision-recall graph was used to evaluate and determine the quality of the results, as the classes are imbalanced. When analysing probabilistic predictions, plots from the charts are generated and used to analyse the performance trade-off for various threshold values.

5.4.1 Vanilla Autoencoder

Table 5-3 compares the flight delay classification and prediction performance based on different input settings regarding area under the precision-recall curve, accuracy, precision and recall. The characteristics were analysed at various model development levels with different input dimensions. The first experiment was evaluated with 3 hidden layers, 33 input dimensions, and 100 epochs, resulting in 42.07% accuracy and 19.99% precision. The hidden layers varied from 3 to 10, with an input dimension from 33 to 500 and epochs from 100 to 300. Increasing the number of epochs and adding one layer led to almost the same results, giving an accuracy of 55.00%, a precision of 27.03% and a recall of 63.57%. It leads to a recall of 55.92%, showing that half of the relevant data were considered. However, no mention of how much unnecessary data were recovered, as seen in the accuracy and precision results.

Table 5-3 Performance comparison of vanilla autoencoder

Hidden	Input	Epochs	Area	Accuracy	Precisi	Recall
10	40	300	0.977	0.8036	0.5246	0.9878
10	100	300	0.946	0.6712	0.3883	0.9037
5	40	200	0.984	0.5651	0.3320	0.9990
5	500	200	0.952	0.7652	0.4691	0.6485
4	33	200	0.852	0.5500	0.2703	0.6357
3	33	100	0.747	0.4207	0.1999	0.5592

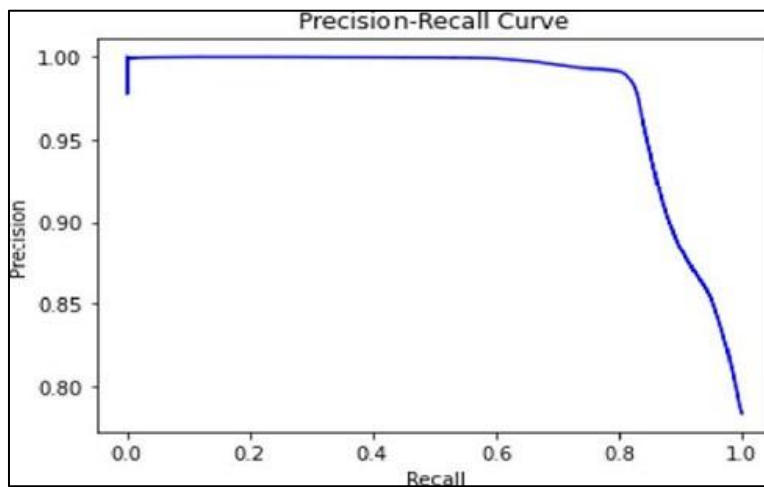
**Figure 5-7 Precision-Recall Curve for 10 Layers and 300 Epochs**

Figure 5-7 represents the precision-recall curve, a plot of precision and recall for different thresholds. The output for the model is constructed with one input layer, 10 hidden layers, 40 input dimensions for each layer, an output layer, which is also the decoder of the autoencoder, and 300 epochs. The activation function used for each hidden layer was Sigmoid. The curve produces the expected form. Precision is high at thresholds with poor recall and diminishes at thresholds with high recall. The high area under the curve shows high recall and precision, highlighting the low false-negative and high false-positive rates, as seen in Figure 5-8. The average score for precision demonstrates that the classifier produces almost accurate results, while the high recall illustrates that the most positive results have been returned.

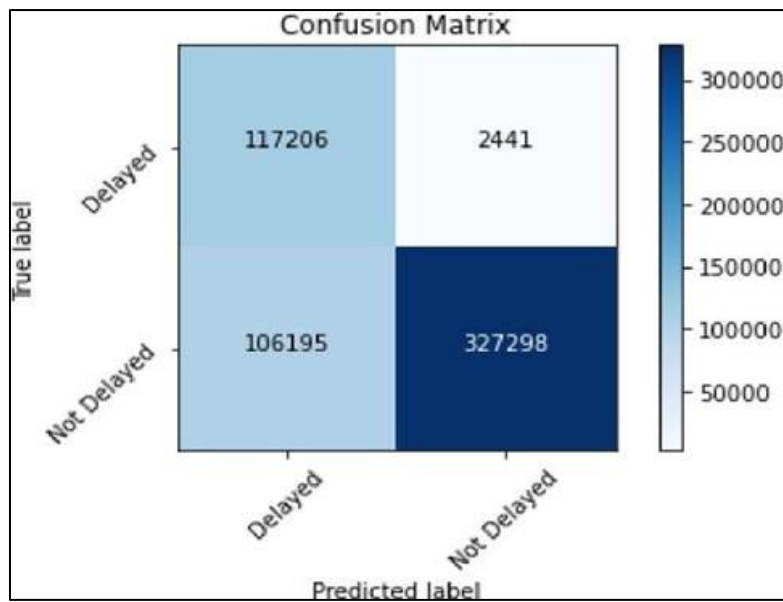


Figure 5-8 Confusion Matrix for 10 Layers and 300 Epochs

Figure 5-8 shows how the model accurately classifies 97.95% of delayed flights and 75.50% of non-delayed aircraft. There are almost 600,000 flight records in this classification. The model accurately identifies 117,206 delayed flights and 327,298 non-delayed flights, which is impressive for the delay prediction software.

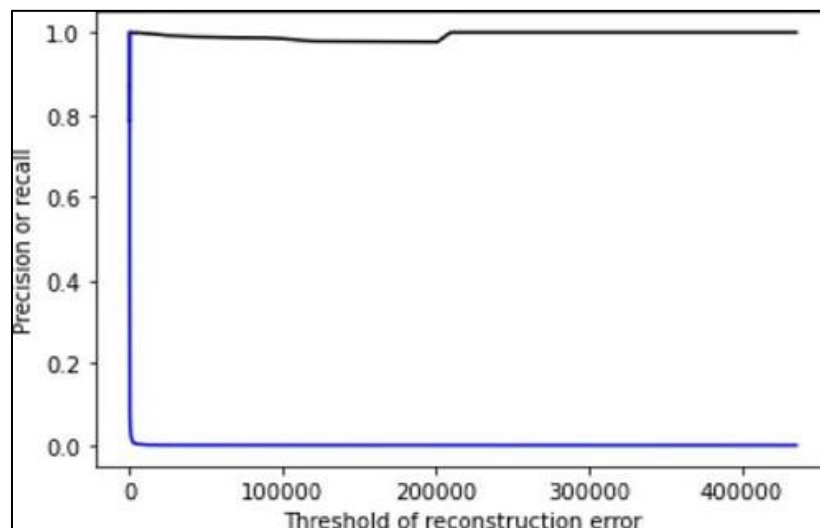


Figure 5-9 Precision and Recall for Different Threshold Values

Figure 5-9 represents the precision and recall resulting from the previous model. The black line represents precision, while recall can be seen in blue. According to the precision equation, reducing the classifier's threshold raises the denominator by expanding the number of outcomes returned. The threshold was set to 0.5 as too high;

the outputs may all be true-positive, increasing precision. If the prior threshold were too low, reducing it further would have generated false positives, reducing precision.

5.4.2 Logistic Regression Autoencoder

Table 5-4 compares the performance of the autoencoder that assembles a logistic regression model built on different input settings in terms of area under the precision-recall curve, accuracy, precision and recall.

Table 5-4 Performance comparison for logistic regression

Hidden	Input	Epochs	Area	Accuracy	Precision	Recall
17	100/50/25	300	0.4184	0.7477	0.7030	0.9784
11	100/50/25/12/	30	0.6384	0.7669	0.7805	0.5536
11	100/50/25/12/	300	0.4229	0.7851	0.7318	0.9936

The characteristics were analysed at various model development levels with different input dimensions. The hidden layers used to implement the autoencoders are 10 to 17, an input dimension of 6 to 100 and epochs of 30 to 300. This implementation uses the same number of hidden layers in the decoder as in the encoder, the last being the output layer, making it an asymmetric structure. The first experiment was evaluated with 11 hidden layers and 30 epochs, resulting in 76.69% accuracy and 78.05% precision. The recall of 55.36% shows that half of the relevant data were considered. However, there is no mention of how much unnecessary data were recovered. The precision and recall graph area was the highest for this implementation.

Increasing the number of epochs by 10 times and adding 6 more layers led to a recall of 97.84%, giving an accuracy of 74.77% and precision of 70.30%. The best result was achieved with 6 hidden layers for the encoder and 5 for the decoder. Inputting 300 epochs, a recall of 99.36% was gained, with an accuracy of 78.51% and a precision of 73.18%.

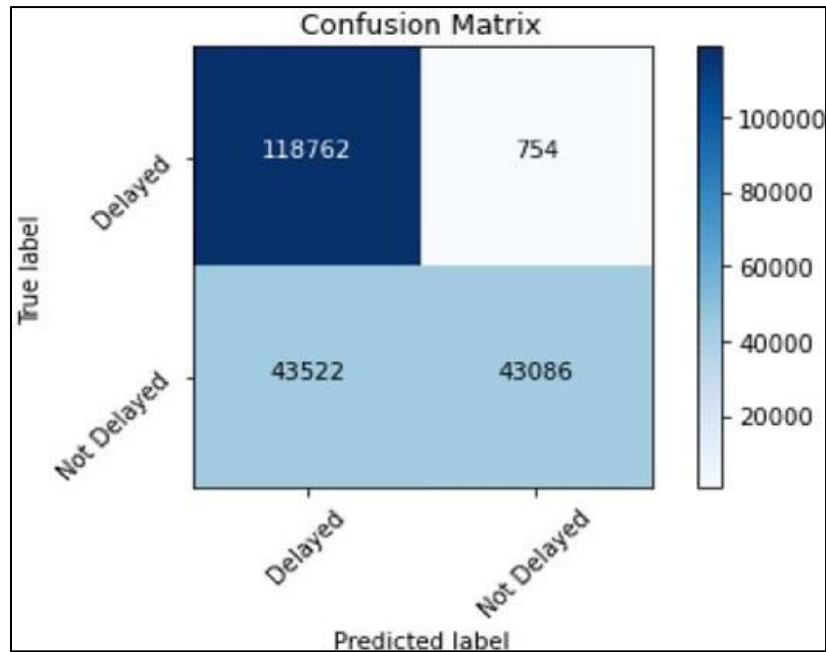


Figure 5-10 Confusion Matrix for 11 Layers and 300 Epochs

Figure 5-10 shows how the model accurately classifies 99.36% of delayed flights and 49.74% of non-delayed aircraft. This classification has almost 200,000 flight records, and the model accurately identifies 118,762 delayed flights and 43,086 non-delayed flights.

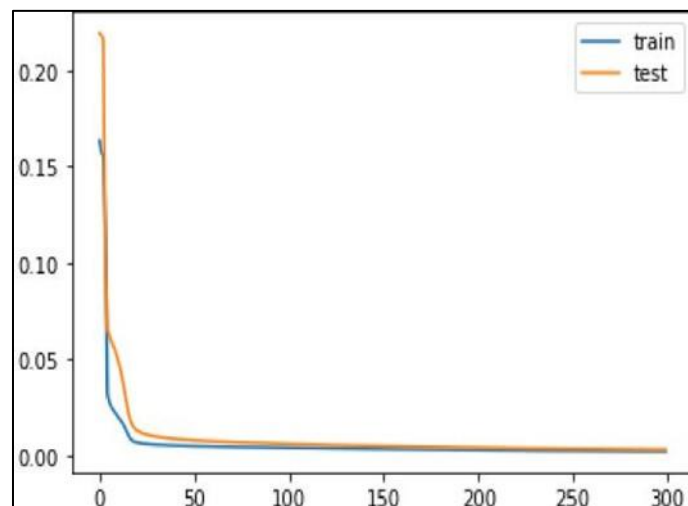


Figure 5-11 Model Loss for 17 Layers and 300 Epochs

Figure 5-11 depicts the logistic regression model's training and test loss. It was found that the model works well during the training and test phases, with consistently great results over the whole training set, which is a major benefit when designing a flight delay classification model.

5.4.3 Multilayer Perceptron Autoencoder

Table 5-5 compares the performance of the multilayer perceptron autoencoder. Here, after fitting the model, the rebuilding component of the model can be neglected, and the model can be used up until the bottleneck. The model produces a fixed-length vector with a compressed version of the input data at the bottleneck. The characteristics were analysed at various model development levels with different input dimensions. The hidden layers used to implement the autoencoders are 5 to 10, with an input dimension of 11, with epochs from 30 to 100. The best result for this case was achieved with 3 hidden layers for the encoder, one hidden layer for the bottleneck and a decoder. Inputting a total of 30 epochs, a recall of 96.19% was gained, with an accuracy of 60.27% and a precision of 59.79%. In this implementation, it can be observed that if the number of epochs increases, the recall begins to drop significantly.

Table 5-5 Performance comparison for multilayer perceptron

Hidden	Input Dimensions	Epoch	Are	Accurac	Precision	Recall
6	11	30	0.4	0.5239	0.5778	0.6662
7	11	30	0.5	0.6027	0.5979	0.9619
7	11	100	0.4	0.5641	0.6320	0.5951
5	11	100	0.4	0.5800	0.5558	0.1821

Figure 5-12 represents the precision-recall curve for the model above. The activation function used for each hidden layer was Sigmoid, while the activation function called 'linear' was used for the output layer. The curve produces the expected form. The area under the curve shows high recall and average precision, highlighting the low false-negative and high false-positive rates. Precision is high at thresholds with poor recall and diminishes at thresholds with high recall.

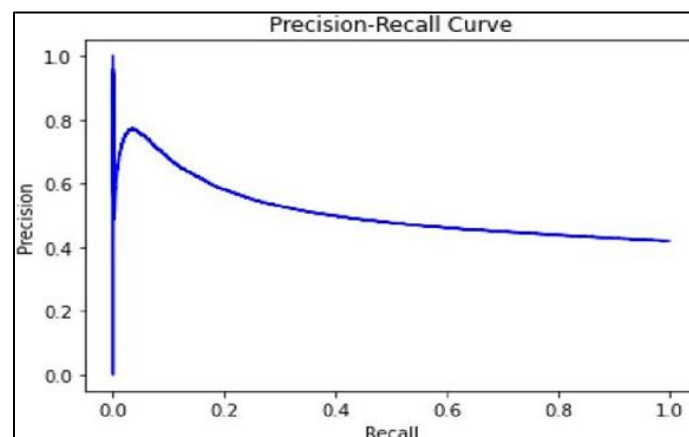


Figure 5-12 Precision-Recall Curve for 7 Layers and 30 Epochs

Figure 5-13 shows how the model accurately classifies 96.19% of delayed flights and 10.65% of non-delayed aircraft. This classification has almost 350,000 flight records, and the model accurately identifies 189,788 delayed flights and 15,222 non-delayed flights.

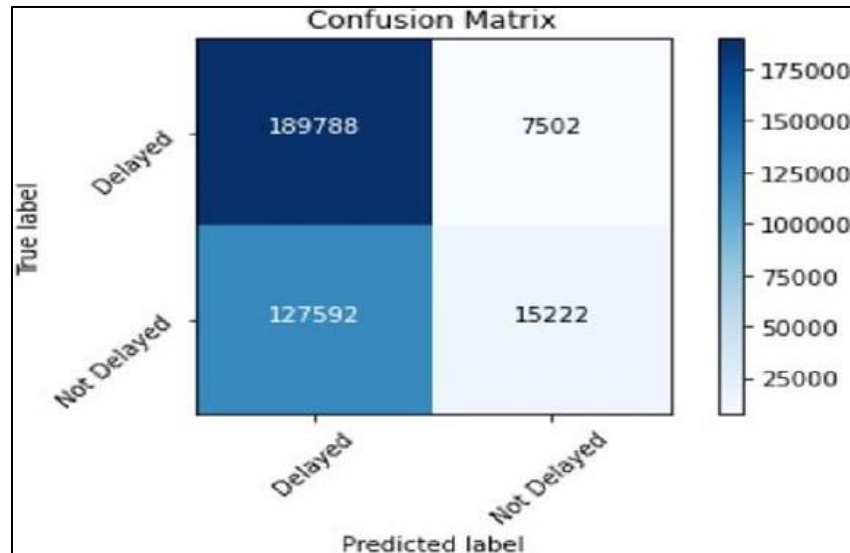


Figure 5-13 Confusion Matrix for 7 Layers and 30 Epochs

5.4.4 Comparative Discussion

In this research, the accuracy, precision and recall for the measures of the Vanilla autoencoder are 80.36%, 52.46% and 98.78%. The output for the logistic regression autoencoder is 78.51% accuracy, 73.18% precision and 99.36% recall, while the multilayer perceptron autoencoder is 60.27%, 59.79%, 96.19%, as shown in Table 5-6. Therefore, the Vanilla autoencoder outperformed the other autoencoders. The main interest is in the autoencoder's output, so the focus was on the other three methods. The multilayer perceptron autoencoder method did not perform as expected.

The advantage of the vanilla autoencoder implementation is that almost 600,000 flight records were analysed in the classification, from which 97.95% of delayed flights and 75.50% of non-delayed flights were classified accurately. The implementation reveals a massive improvement when using a deep autoencoder, such as vanilla, having at least 200,000 more flight records analysed in the classification and successfully classifying over 25% of non-delayed aircraft compared to the logistic regression model. A method with numerous parameters to fit during training can produce different outcomes in classification tasks than other methods. The area gives another aspect where the vanilla autoencoder outperforms the other methods under

the precision-recall curve. The wide area of 0.9773 below the curve denotes good recall and precision, with high precision implying a low false-positive rate and high recall indicating a low false-negative rate, compared to 0.4229 for logistic regression and 0.5121 for multilayer perceptron autoencoder.

Logistic regression autoencoder comes into an advantage with the loss function analysis. The experiment revealed a massive improvement from the precision point of view, at least 20% compared to the other methods. It was found that the model works well during the training and test phases, with consistently great results over the whole training set, which is a major benefit when designing a flight delay classification model.

Table 5-6 Performance comparisons of the different autoencoders

Model	Accuracy	Precision	Recall
Vanilla	0.8036	0.5246	0.987
Logistic regression	0.7851	0.7318	0.993
multilayer perceptron	0.6027	0.5979	0.961

5.5 Conclusion and Future Direction

Predicting flight delays is a fascinating study issue that has gotten considerable attention recently. Most studies have attempted to create and expand their models to improve the accuracy and precision of flight delay prediction. Because on-time flights are critical, flight delay prediction algorithms must be exact and accurate. In this study, we proposed implementing three approaches to an autoencoder: vanilla autoencoder, logistic regression autoencoder, and multilayer perceptron autoencoder. We compared the deep feedforward neural network's predicted accuracy, precision and recall with the autoencoder approaches at different variable tunings and epochs. Our primary aim was to classify and predict flight delays using an autoencoder and re-scale its parameters.

Our experimental results show that the deep vanilla autoencoder outperformed the other two implementations at different parameter adjustments. Comparing the three autoencoder models shows that the vanilla autoencoder's accuracy is greater by 1.85% than the logistic regression autoencoder and by 20.09% than the multilayer perceptron autoencoder. At the same time, the area under the precision-recall curve is higher by 0.5744 and 0.4852; using the vanilla autoencoder to optimise the results positively affects classifying and predicting the delay forecasting.

Flight delays are a popular issue because of their economic and environmental consequences. They may raise customer costs as well as airline operating costs. Aside from direct passenger effects, delay prediction is critical for every stakeholder in the air transport industry during the decision-making phase. This work will assist the aviation industry, especially the air transportation sector, enhance passenger experiences by improving flight delay decision support. The next step would be to test the correctness of this method on different datasets or sample data.

Acknowledgement & Funding

The Petroleum Trust Development Fund (PTDF) Nigeria partially funded this work through grant number PTDF/ED/OSS/PHD/DBB/1558/19 for one of the first author's PhD studies. We also thank the UKRI for the Covid-19 recovery grant under the budget code SA077N.

CRedit Authorship Contribution Statement

Desmond Bala Bisandu, Dan A. Soviani-Sitoiu: Conceptualisation, Methodology, Investigation, Data curation Writing – original draft. Irene Moulitsas: Supervision, Project administration, Funding acquisition, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data Availability

All data used in this research was obtained from the United States Bureau of Transportation Statistics website.

5.6 References

1. Tan YE., Teong KS., Shabbir M., Foo LK., Chua SL. Modelling flight delays in the presence of class imbalance. *ACM International Conference Proceeding Series*. 2018;; 186–191. Available at: DOI:10.1145/3299819.3299847
2. Carvalho L., Sternberg A., Maia Gonçalves L., Beatriz Cruz A., Soares JA., Brandão D., et al. On the relevance of data science for flight delay research: a systematic review. *Transport Reviews*. Taylor & Francis; 2021; 41(4): 499–528. Available at: DOI:10.1080/01441647.2020.1861123
3. Mofokeng TJ., Marnewick A. Factors contributing to delays regarding aircraft during A-check maintenance. 2017 IEEE Technology and Engineering Management Society Conference, TEMSCON 2017. 2017. pp. 185–190. Available at: DOI:10.1109/TEMSCON.2017.7998375
4. Dhanawade R., Deo M., Khanna N., Deolekar R V. Analysing factors influencing flight delay prediction. *Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development, INDIACOM 2019*. 2019; : 1003–1007.
5. Mizufune K., Katsumata S. Joint classification model of topic and polarity: Finding satisfaction and dissatisfaction factors from airport service review. *IEEE International Conference on Data Mining Workshops, ICDMW*. IEEE; 2019; 2018-Novem: 856–863. Available at: DOI:10.1109/ICDMW.2018.00126
6. Fernandes N., Moro S., Costa CJ., Aparício M. Factors influencing charter flight departure delay. *Research in Transportation Business and Management*. Elsevier; 2020; 34(December): 100413. Available at: DOI:10.1016/j.rtbm.2019.100413
7. Balakrishna P., Ganesan R., Sherry L. Accuracy of reinforcement learning algorithms for predicting aircraft taxi-out times: A case-study of Tampa Bay departures. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2010; 18(6): 950–962. Available at: DOI:10.1016/j.trc.2010.03.003
8. Yu B., Guo Z., Asian S., Wang H., Chen G. Flight delay prediction for commercial air transport: A Deep Learning approach. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier; 2019; 125(March): 203–221. Available at: DOI:10.1016/j.tre.2019.03.013
9. Ye B., Liu B., Tian Y., Wan L. A methodology for predicting aggregate flight departure delays in airports based on supervised learning. *Sustainability*. 2020; 12(7): 2749. Available at: DOI:10.3390/su12072749
10. Gupta H. Evaluating service quality of airline industry using hybrid best worst method and VIKOR. *Journal of Air Transport Management*. Elsevier Ltd; 2018; 68: 35–47. Available at: DOI:10.1016/j.jairtraman.2017.06.001
11. Hondet G., Delgado L., Gurtner G. Airline disruption management with aircraft swapping and reinforcement learning. *SESAR Innovation Days*. 2018;
12. Pejovic T., Noland RB., Williams V., Toumi R. A tentative analysis of the impacts of an airport closure. *Journal of Air Transport Management*. Elsevier Ltd; 2009; 15(5): 241–248. Available at: DOI:10.1016/j.jairtraman.2009.02.004

13. Bhadra D. You (expect to) get what you pay for: A system approach to delay, fare, and complaints. *Transportation Research Part A: Policy and Practice*. Elsevier Ltd; 2009; 43(9–10): 829–843. Available at: DOI:10.1016/j.tra.2009.08.003
14. Shabanpour N., Razavi-Termeh SV., Sadeghi-Niaraki A., Choi SM., Abuhmed T. Integration of Machine Learning algorithms and GIS-based approaches to cutaneous leishmaniasis prevalence risk mapping. *International Journal of Applied Earth Observation and Geoinformation*. Elsevier B.V.; 2022; 112(June): 102854. Available at: DOI:10.1016/j.jag.2022.102854
15. D'Ariano A., Pistelli M., Pacciarelli D. Aircraft retiming and rerouting in vicinity of airports. *IET Intelligent Transport Systems*. 2012; 6(4): 433–443. Available at: DOI:10.1049/iet-its.2011.0182
16. Campanelli B., Fleurquin P., Eguíluz VM., Ramasco JJ., Arranz A., Extebarria I., et al. Modelling reactionary delays in the European air transport network. *SIDs 2014 - Proceedings of the SESAR Innovation Days*. 2014.
17. Ogunsina K., Okolo WA. Artificial Neural Network Modelling for Airline Disruption Management. 2021; Available at: DOI:10.2514/1.i011018
18. Chen M., Zeng W., Xu Z., Li J. Delay prediction based on deep stacked autoencoder networks. *ACM International Conference Proceeding Series*. 2018; Part F1482: 238–242. Available at: DOI:10.1145/3321619.3321669
19. Yazdi MF., Kamel SR., Chabok SJM., Kheirabadi M. Flight delay prediction based on Deep Learning and Levenberg-Marquart algorithm. *Journal of Data*. Springer International Publishing; 2020; 7(1). Available at: DOI:10.1186/s40537-020-00380-z
20. Esmaeilzadeh E., Mokhtarimousavi S. Machine Learning approach for flight departure delay prediction and analysis. *Transportation Research Record*. 2020; 2674(8): 145–159. Available at: DOI:10.1177/0361198120930014
21. Ciruelos C., Arranz A., Extebarria I., Peces S., Campanelli B., Fleurquin P., et al. Modelling delay propagation trees for scheduled flights. *Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2015*. 2015. pp. 1–10. Available at: https://www.researchgate.net/profile/Victor-Eguiluz/publication/336837610_Modelling_Delay_Propagation_Trees_for_Scheduled_Flights/links/5db5bbe792851c577ecebb30/Modelling-Delay-Propagation-Trees-for-Scheduled-Flights.pdf
22. Rong F., Qianya L., Bo H., Jing Z., Dongdong Y. The prediction of flight delays based the analysis of Random flight points. *Chinese Control Conference, CCC*. 2015; 2015-Sept: 3992–3997. Available at: DOI:10.1109/ChiCC.2015.7260255
23. Kim YJ., Choi S., Briceno S., Mavris D. A Deep Learning approach to flight delay prediction. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. IEEE; 2016; 2016-Decem: 1–6. Available at: DOI:10.1109/DASC.2016.7778092
24. Cai KQ., Zhang J., Xiao MM., Tang K., Du WB. Simultaneous optimisation of airspace congestion and flight delay in air traffic network flow

- management. *IEEE Transactions on Intelligent Transportation Systems*. 2017; 18(11): 3072–3082. Available at: DOI:10.1109/TITS.2017.2673247
25. Proenca HM., Klijn R., Bäck T., Van Leeuwen M. Identifying flight delay patterns using diverse subgroup discovery. *Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence, SSCI 2018*. 2019. pp. 60–67. Available at: DOI:10.1109/SSCI.2018.8628933
 26. Manna S., Biswas S., Kundu R., Rakshit S., Gupta P., Barman S. A statistical approach to predict flight delay using gradient boosted decision tree. *ICCIDS 2017 - International Conference on Computational Intelligence in Data Science, Proceedings*. 2018; 2018-Janua: 1–5. Available at: DOI:10.1109/ICCIDS.2017.8272656
 27. Venkatesh V., Arya A., Agarwal P., Lakshmi S., Balana S. Iterative machine and Deep Learning approach for aviation delay prediction. *2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics, UPCON 2017*. 2017. Available at: DOI:10.1109/UPCON.2017.8251111
 28. Bisandu DB., Homaid MS., Moulitsas I., Filippone S. A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective. *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK. ACM; 2021*. pp. 1–10. Available at: DOI:https://doi.org/10.1145/3505711.3505712
 29. Choi S., Kim YJ., Briceno S., Mavris D. Cost-sensitive prediction of airline delays using Machine Learning. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. 2017; 2017-Sept. Available at: DOI:10.1109/DASC.2017.8102035
 30. Hu C., Wei H., B MC. Enhancing Airlines Delay Prediction by Implementing Classification Based Deep Learning Algorithms. *Springer International Publishing; 2019*. 563–581 p. Available at: DOI:10.1007/978-3-030-19063-7
 31. Rebollo JJ., Balakrishnan H. Characterization and prediction of air traffic delays. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2014; 44: 231–241. Available at: DOI:10.1016/j.trc.2014.04.007
 32. Balakrishna P., Ganesan R., Sherry L., Levy BS. Estimating taxi-out times with a reinforcement learning algorithm. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. 2008; : 1–12. Available at: DOI:10.1109/DASC.2008.4702812
 33. Yang C., Marshall ZA., Mott JH. A novel integration platform to reduce flight delays in the National Airspace System. *2020 Systems and Information Engineering Design Symposium, SIEDS 2020*. 2020; : 1–6. Available at: DOI:10.1109/SIEDS49339.2020.9106657
 34. Bisandu DB., Moulitsas I., Filippone S. Social ski driver conditional autoregressive-based Deep Learning classifier for flight delay prediction. *Neural Computing and Applications*. Springer London; 2022; 34(11): 8777–8802. Available at: DOI:10.1007/s00521-022-06898-y
 35. Triantafillou E., Zhu T., Dumoulin V., Lamblin P., Evci U., Xu K., et al. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. *ICLR*. 2020. pp. 1–24. Available at: <http://arxiv.org/abs/1903.03096>

36. Ionescu L., Gwiggner C., Kliever N. Data Analysis of Delays in Airline Networks. *Business and Information Systems Engineering*. Springer Fachmedien Wiesbaden; 2016; 58(2): 119–133. Available at: DOI:10.1007/s12599-015-0391-3
37. Pham DT., Trant NP., Goh SK., Alam S., Duong V. Reinforcement learning for two-aircraft conflict resolution in the presence of uncertainty. *RIVF 2019 - Proceedings: 2019 IEEE-RIVF International Conference on Computing and Communication Technologies*. 2019; Available at: DOI:10.1109/RIVF.2019.8713624
38. Mangortey E., Pinon-Fischer OJ., Puranik TG., Mavris DN. Predicting The Occurrence of Weather And Volume Related Ground Delay Programs. 2019; (June): 1–29. Available at: DOI:10.2514/6.2019-3188
39. Heer J., Kong N., Agrawala M. Sizing the horizon. 2009; : 1303. Available at: DOI:10.1145/1518701.1518897
40. Gu Y., Gowda GM., Jayanna PK., Boumghar R., Lucas L., Bernardi A., et al. The added value of advanced feature engineering and selection for Machine Learning models for spacecraft behavior prediction. *15th International Conference on Space Operations*, 2018. 2018; (June): 1–11. Available at: DOI:10.2514/6.2018-2561
41. Campanelli B., Fleurquin P., Arranz A., Etxebarria I., Ciruelos C., Eguíluz VM., et al. Comparing the modelling of delay propagation in the US and European air traffic networks. *Journal of Air Transport Management*. 2016; 56(Part A): 12–18. Available at: DOI:10.1016/j.jairtraman.2016.03.017
42. Hu J., Zhang J., Zhang C., Wang J. A new deep neural network based on a stack of single-hidden-layer feedforward neural networks with randomly fixed hidden neurons. *Neurocomputing*. Elsevier; 2016; 171: 63–72. Available at: DOI:10.1016/j.neucom.2015.06.017
43. Yi J., Zhang H., Liu H., Zhong G., Li G. Flight Delay Classification Prediction Based on Stacking Algorithm. *Journal of Advanced Transportation*. 2021; 2021. Available at: DOI:10.1155/2021/4292778
44. Samatin Njikam AN., Zhao H. A novel activation function for multilayer feedforward neural networks. *Applied Intelligence*. Applied Intelligence; 2016; 45(1): 75–82. Available at: DOI:10.1007/s10489-015-0744-0
45. Nwankpa CE., Ijomah W., Gachagan A., Marshall S. Activation functions: Comparison of trends in practice and research for Deep Learning. *arXiv*. 2018; : 1–20.
46. Orimoloye LO., Sung MC., Ma T., Johnson JEV. Comparing the effectiveness of deep feedforward neural networks and shallow architectures for predicting stock price indices. *Expert Systems with Applications*. Elsevier Ltd; 2020; 139: 112828. Available at: DOI:10.1016/j.eswa.2019.112828
47. Arshad M., Ahmed M. Prediction of Train Delay in Indian Railways through Machine Learning Techniques. *International Journal of Computer Sciences and Engineering*. 2019; 7(2): 405–411. Available at: DOI:10.26438/ijcse/v7i2.405411
48. Lillicrap TP., Hunt JJ., Pritzel A., Heess N., Erez T., Tassa Y., et al. Continuous control with deep reinforcement learning. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track*

- Proceedings. 2016;
49. Ni W., Cassidy MJ. Cordon control with spatially-varying metering rates: A Reinforcement Learning approach. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(September 2018): 358–369. Available at: DOI:10.1016/j.trc.2018.12.007
 50. George E., Khan SS. Reinforcement learning for taxi-out time prediction: An improved Q-learning approach. 2015 International Conference on Computing and Network Communications, CoCoNet 2015. IEEE; 2016; : 757–764. Available at: DOI:10.1109/CoCoNet.2015.7411275
 51. Sharifzadeh S., Chiotellis I., Triebel R., Cremers D. Learning to Drive using Inverse Reinforcement Learning and Deep Q-Networks. 2016; (Nips): 1–7. Available at: <http://arxiv.org/abs/1612.03653>
 52. Alla H., Moumoun L., Balouki Y. A Multilayer Perceptron Neural Network with Selective-Data Training for Flight Arrival Delay Prediction. *Scientific Programming*. 2021; 2021. Available at: DOI:10.1155/2021/5558918
 53. Musaddi R., Jaiswal A., Girdonia M. Flight Delay Prediction using Binary Classification. *International Journal of Emerging Technologies in Engineering Research (IJETER)*. 2018; 6(May). Available at: www.ijeter.everscience.org
 54. Kim YJ. A DEEP LEARNING AND PARALLEL SIMULATION METHODOLOGY FOR AIR TRAFFIC MANAGEMENT A Dissertation Presented to The Academic Faculty. 2017; (December). Available at: <https://smartech.gatech.edu/bitstream/handle/1853/59180/KIM-DISSERTATION-2017.pdf?sequence=1&isAllowed=y>
 55. Hans RT. Using a biometric system to check-in and check-out luggage at airports. *Proceedings - 2014 International Symposium on Biometrics and Security Technologies, ISBAST 2014*. IEEE; 2014. pp. 63–66. Available at: DOI:10.1109/ISBAST.2014.7013095
 56. Fei T., De Joux N., Kefalidou G., D'cruz M., Sharples S. Towards understanding information needs and user acceptance of mobile technologies to improve passenger experience in airports. *ECCE '16: Proceedings of the European Conference on Cognitive Ergonomics*. 2016. Available at: DOI:10.1145/2970930.2970938
 57. Reitman S., Alam S., Schultz M. Advanced Quantification of Weather Impact on Air Traffic Management. 13th USA/Europe Air Traffic Management Research and Development Seminar 2019. 2019;
 58. Weigang L., Alves CJP., Omar N. An expert system for air traffic flow management. *Journal of Advanced Transportation*. 1997; 31(3): 343–361. Available at: DOI:10.1002/atr.5670310308
 59. Prentice C., Kadan M. The role of airport service quality in airport and destination choice. *Journal of Retailing and Consumer Services*. Elsevier Ltd; 2019; 47(July 2018): 40–48. Available at: DOI:10.1016/j.jretconser.2018.10.006
 60. Bosson CS., Nikoleris T. Supervised learning applied to air traffic trajectory classification. *AIAA Information Systems-AIAA Infotech at Aerospace*, 2018. 2018; (209989). Available at: DOI:10.2514/6.2018-1637
 61. Wang Y. Prediction of weather impacted airport capacity using ensemble learning. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*.

- IEEE; 2011; : 1–11. Available at: DOI:10.1109/DASC.2011.6096002
62. Zhang Y., Gao J., Zhou H. Breeds Classification with Deep Convolutional Neural Network. ACM International Conference Proceeding Series. 2020; : 145–151. Available at: DOI:10.1145/3383972.3383975
 63. Irmak H., Ömür T., Tosun N. Flight delay prediction based with Machine Learning. Scientific Journal of Logistics. 2022; 18(1): 97–107. Available at: DOI:http://doi.org/10.17270/J.LOG.2021.655
 64. Dou X. Flight Arrival Delay Prediction And Analysis Using Ensemble Learning. 2020; (It nec): 836–840. Available at: DOI:10.1109/itnec48623.2020.9084929
 65. Korvesis P., Besseau S., Vazirgiannis M. Predictive maintenance in aviation: Failure prediction from post-flight reports. Proceedings - IEEE 34th International Conference on Data Engineering, ICDE 2018. IEEE; 2018; : 1423–1434. Available at: DOI:10.1109/ICDE.2018.00160
 66. Yao S. Emerging Technologies in Data Mining and Information Yao, S. (2019). Emerging Technologies in Data Mining and Information Security. In Emerging Technologies in Data Mining and Information Security (Vol. 813). <https://doi.org/10.1007/978-981-13-1498-8Secur>. Emerging Technologies in Data Mining and Information Security. Springer Singapore; 2019. 871–877 p. Available at: DOI:10.1007/978-981-13-1498-8
 67. Bian Y., Yang J., Li M., Lan R. Automated flare prediction using extreme learning machine. Mathematical Problems in Engineering. 2013; 2013. Available at: DOI:10.1155/2013/917139
 68. Stefanovič P., Štrimaitis R., Kurasova O. Prediction of Flight Time Deviation for Lithuanian Airports Using Supervised Machine Learning Model. Computational Intelligence and Neuroscience. 2020; 2020. Available at: DOI:10.1155/2020/8878681

Chapter 6: A Hybrid Ensemble Machine Learning Approach for Arrival Flight Delay Classification Prediction Using Voting Aggregation Technique

The number of flights keeps increasing with the development of civil aviation, and flight delays have become a serious issue of concern to the aviation industry. The need for reliable flight classification and prediction cannot be overemphasised because of its importance. In this research, we perform arrival flight delay using a hybrid ensemble Machine Learning algorithm based on voting aggregation. Each ensemble's architecture consists of four supervised Machine Learning algorithms: Logistic Regression, Decision Tree, Support Vector Classifier and Random Forest. We conducted a comparative experiment based on the United States Bureau of Statistics dataset to verify the efficacy of our proposed approach against other benchmark approaches. We employ multiple evaluation metrics to check the performance of our proposed model comprehensively: Accuracy, Recall, Precision, F1-Score, AUC Score and ROC curve. Our experimental results show that our model has better prediction accuracy and confusion matrix on the binary classification of flight delays task. Hence, the hybrid ensemble model presented in this research can be used as a decision-support model to improve passenger travel itineraries.

6.1 Introduction

Many factors, such as safety, security, air carrier, maintenance, National Aviation System (NAS), weather and airport scheduling, affect flight plans in the civil aviation transportation processes [1-4]. Frequently, scheduled flights cannot arrive on time, affecting subsequent flights. Flight delay occurs conventionally if the flight is 15 minutes later than the scheduled departure or arrival time. Air traffic demand-capacity imbalances in arrival/departure flights resulting from continuous air traffic growth with limited airport expansion capabilities. For example, in 2017, there was 4.3% on European flights relative to 2016 [3]. It corresponds to an average of 1191 flights per day. Also, 20.4% of flights in 2017 experienced arrival delays of 15 minutes and above [6-8]. Predicting flight delays to improve airport efficiency and operation coordination is important. These delays have caused more damage to the aviation industry and passengers with almost an exponential increment, leading to poor passenger satisfaction and a loss of at least \$20.5 million to flight delay-related issues in the United States in 2018 [5, 9,10].

With the rapid increase in data from the air transportation industry due to the constant development in the sector, processing a huge amount of data is becoming tedious and almost impractical with only traditional processing methods such as balancing and shuffling. Flight delay prediction based on Machine Learning has performed well in the past. Some methods such as KNN, decision tree and random forest [3], Support vector classifier, and others have good accuracy on large flight on-time datasets but are slow in computation compared with Deep Learning methods. Machine Learning superior performance in prediction compared with other traditional statistical methods has been massively recorded in Artificial Intelligence and transportation-related research [11,12,13,14,15]. Support Vector Machine (SVM) is among the most popular Machine Learning techniques for classification problems because of high prediction results, especially with high dimensional and imbalanced datasets [13,16,17]. However, due to the high dimension of the feature, good pre-processing is required to perform flight delay predictions. For instance, one-hot encoding processing will be needed for flight landings and take-off features, leading to sparse data. Encoding aircraft tail coding and airlines may also be required. This encoding process removed much information that may affect the model's performance output.

In recent years, machine and Deep Learning have become the most popular and acceptable methods to predict flight delays [8,18]. Many researchers have proposed techniques for flight delay predictions using multiple perspectives and uncertainty factors affecting the flight's operation. For instance, using a cause-and-effect test, Wen-BoDu [18] establishes a delay cause-and-effect network (DCN) by analysing the factors that affect flight delays. Alice Sternberg [19] studied hidden flight delays using association rules. Also, Bin Yu [16] used a deep belief network to analyse and predict internal flight delay patterns. Poornima [17] applied a non-parametric reinforcement learning method (RL) to study the actual time of aircraft and taxi time deviation. Rodriguez-San et al. [18] and Cheng-Lung Wu et al. [19] use Bayesian network and reliability models to predict flight delays. Roberto Henriques [20] uses a multilayer perceptron to predict flight delay. Sun Choi et al. [21] built a model for flight delay prediction based on Ada-boost with a sample imbalance flight data set. The result of the forecast was relatively good.

Ensemble learning methods, such as the Boosting model, have improved the accuracy of weak classification algorithms. A Series of prediction functions are

constructed to reduce the loss continuously through a series of iterations by the model. Cat-boost is an open-source algorithm for gradient enhancement developed by Dorogush [22]. The users are processing classification characteristics from a large dataset, which can solve a ranking problem, classification and regression, especially in the imbalance dataset prediction, which has superior performance than the XG-boost and Light-GBM as shown in previous studies [22, 23]. Cat-boost applies the symmetric tree method in processing categorical features, which has an advantage over XG-boost in improving the model's performance and robustness with a high accuracy rate.

Our contribution is summarised as follows: Firstly, we proposed a hybrid ensemble learning model based on hard voting techniques for predicting arrival flight delay with a feature sub-sampling technique. Secondly, our proposed methodology generalisation relies on a relative performance comparison between the strategic schedule of flight delay prediction on different well-known ensemble machine learners. Thirdly, we provide experimental evidence of our proposed method results compared with Boosting and stacking approaches. We clearly show the improvement in our proposed method.

The rest of the paper's organisation is as follows. Section 6.2 presents the introduction. Section 6.3 contains an overview of some most important related works. The materials and methods are presented in Section 6.4. Section 6.5 presents the results and discussion. Finally, Section 6.6 presents the conclusion.

6.2 Related Works

Any disruption in the air transportation system processes, aside from the Traffic Flow Management (TFMs), including but not limited to a mechanical problem, severe weather conditions, an outage at the origin/destination airports, or staff shortage, may result in a flight delay. Air transportation delays may occur at different flight phases: en-route, arrival and departure and can affect passenger schedules, crew and flight (e.g., missing a connecting flight). Also, delay propagation can affect arrival and departure flights [23, 24]. Therefore, the ability to control the factors and predict flight delays is an essential objective for airlines.

Flight delay analysis and prediction have been addressed extensively in the literature, as seen in [25, 26, 27]. A data-driven model for flight delay distribution estimation was proposed by [25]. Flight statistics were determined for ten major US

airports through a historical dataset. Departure and arrival delays have been modelled based on these statistics as normal and Poisson processes [28]. The historical data from Beijing Capital International Airport was used to estimate the probability density function of departure and arrival delays of flights with an optimal generalised extreme value model [27]. The authors in [29] proposed a statistical model that estimates departure flight delay distributions and seasonal patterns using the Denver International Airport. Random residuals' seasonal and daily propagation patterns were considered. In [30], a flight delay detection pattern based on a frequency analysis at Orlando International Airport was proposed to analyse airport network propagation delays. The authors used a Bayesian network for the network queue [31].

Authors in [32, 33] proposed Machine Learning approaches for flight delay classification with a prediction horizon of several days before the flight execution occurs. In [32], the authors achieved an accuracy rate of 0.268 with five days of weather forecast available before the flight's execution. A random forest classifier with exclusively weather-related feature training was employed. Authors in [33] used unfavourable weather conditions to exclusively propose a flight delay classification model. A balanced dataset was used with a random sampling algorithm to decrease delayed samples based on the under-sampling algorithm. The features considered were the origin/destination airport, weather conditions, and scheduled departure/arrival time. An accuracy rate of 0.86 is achieved with a recall of 0.87 on a threshold of 60 minutes (i.e., any flight with 60 minutes or more time relative to arrival time is considered delayed) by the random forest method. A Receiver Operating Characteristics (ROC) of XGBoost achieved was 0.53 for prediction in 20 Asian airports for a low-cost airline. From the literature results, ensemble methods can generate robust results for classifying datasets with noise having an improved prediction than a single classifier [34, 35, 36].

Additionally, ensemble learners have many advantages; multi-classifier systems have proven to be the potential for developing robust learning methods with noisy data in both class labels, missing values and features. Classifier results can have increased variance due to noisy training data. However, the variance can be reduced by learning the combined decision of a committee of hypotheses. In particular, bagging has shown outstandingly high noise levels as a variance-reducing method.

Despite all the efforts to find universal solutions to the problem of flight delay prediction through the various contributions from academia and the industries, It has been reiterated that there is a need to develop further hybrid models that can effectively and efficiently apply a repeatable and systematic approach to identify and predict flight delays from different viewpoints. Identifying and measuring the factors that significantly affect flight delays is important to the aviation community and cannot be overemphasised. Because it will help controllers, traffic managers, and airlines proactively adjust and minimise operational impacts. Also, identifying each factor's robust occurrence probability in the delay scenarios will guide airlines and other stakeholders in the planning process and disruption recovery. Thus, we proposed a hybrid ensemble Machine Learning method based on the voting aggregation technique for arrival flight delays for non-weather impacted flight delays.

6.3 Materials and Methods

In this research, we proposed an ensemble members construction approach by learning from the sub-set samples of features. The proposed method is shown in Figure 6-1 and can be summarised in the following stages:

Stage 1: Feature sub-sampling: The sample features will produce accurate and diverse classifiers. A hybrid sub-sampling method is applied for the feature sub-sampling.

Stage 2: Candidate classifiers generation: Machine Learning methods are applied to generate a pool of candidate classifiers. The input for each classifier is a subset of the features generated from Stage 1. Support Vector Classifier (SVC) is chosen as the Machine Learning algorithm for this study's base classifier. SVC application is due to its ability to deal with high dimensional training data features.

Stage 3: Ensemble Committee Construction: Two sub-steps are involved. The first is the base classifier behaviour characterisation generated. The other is fusing the voting aggregation of the classifiers that construct a robust committee for classification. The high diversity of the classifiers means classifiers disagree with each other but have good classification accuracy as committee members. The majority voting mechanism is then applied to make the final decision for the classification ensemble.

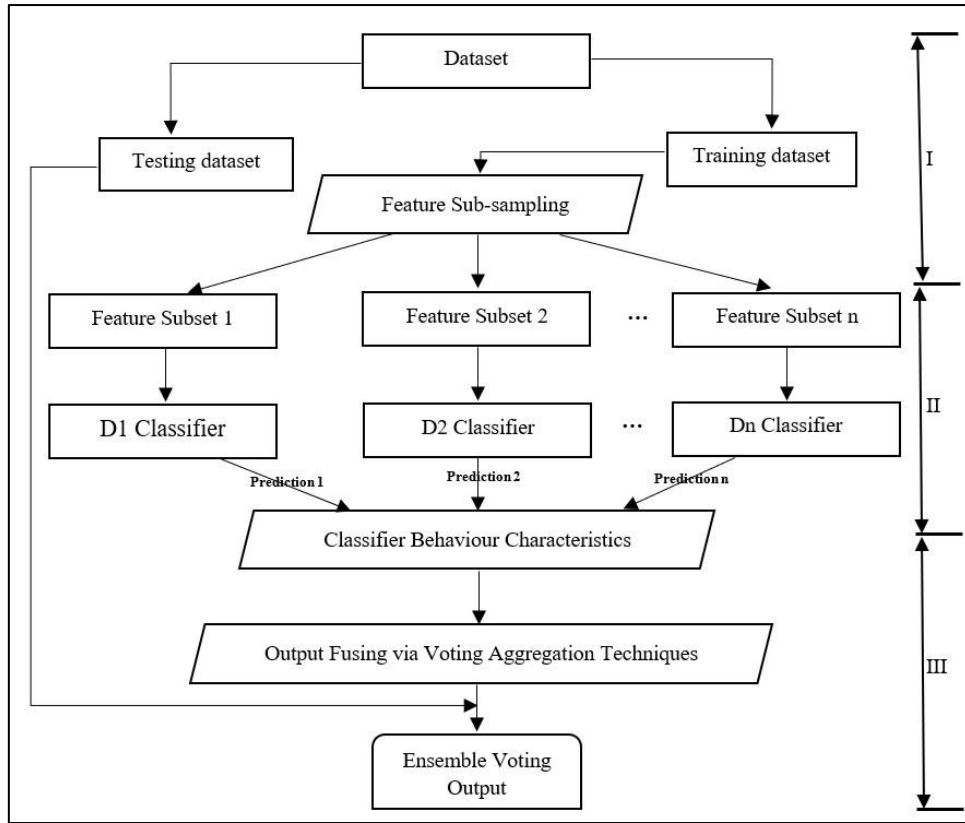


Figure 6-1 The Proposed Ensemble Method

6.3.1 Feature Characterisation

We utilised a hybrid feature selection algorithm to select features to help produce accurate and diverse classifiers. Our approach uses a random selection of features to construct the base classifiers from the original dataset. The randomise selection method makes the ensemble classification model less sensitive to feature selection.

Assuming a dataset for flight on time is represented by a matrix M is $m \times n$ and $M = [f_{ij}]_{m \times n}$ where m is the number of learning samples, and n is the number of features. An instance (Matrix M row) is an expression of a vector $X_i = (f_{i1}, f_{i2}, \dots, f_{in})$, where f_{ij} represents the index of i th learning sample in the j th feature. Every sample is labelled with a class $c_j = \{-1, +1\}$. Assuming we have learning instances from $j = 1$ to k for class $+1$ and learning samples from $j = k + 1$ to n are for class -1 , the i th feature vectors expression for class $+1$ and -1 are expressed as $X(i, +) = \{f_{i1}, f_{i2}, \dots, f_{in}\}$ and $X(i, -) = \{f_{i(k+1)}, f_{i(k+2)}, \dots, f_{in}\}$, respectively.

The feature's significance is important because it can affect the performance of the individual base classifiers and is measured by the features' discrimination power. Characterisation of the feature's discrimination power measures the difference between the different groups' expression levels of learning samples [37]. Equation (6-1) shows the discriminant power.

$$d(i) = \frac{\text{diff}(X(i,+), X(i,-1))}{l_0 + l}, \quad (6-1)$$

Where $\text{diff}(X(i,+), X(i,-1))$ is the group difference between $X(i,+)$ and $X(i,-)$, l_0 is a regularisation constant, and s is the standard error.

Measuring features' discrimination power is presented in [38] to select desired features for classification. For a feature in the i th, let $[\mu_+(i), \sigma_+(i)]$ and $[\mu_-(i), \sigma_-(i)]$ represents the means and standard deviation of the learning sample expression in class $+1$ and -1 , respectively. As shown in Equation (6-2) and Equation (6-3), Equation (6-4) and Equation (6-5), respectively.

$$\mu_+(i) = \frac{1}{k} \sum_{j=1}^k f_{ij}, \quad (6-2)$$

$$\sigma_+(i) = \sqrt{\frac{\sum_{j=1}^k (f_{ij} - \mu_+(i))^2}{k}} \quad (6-3)$$

and

$$\mu_-(i) = \frac{1}{n-k} \sum_{j=k+1}^n f_{ij}, \quad (6-4)$$

$$\sigma_-(i) = \sqrt{\frac{\sum_{j=k+1}^n (f_{ij} - \mu_-(i))^2}{n-k}} \quad (6-5)$$

The measure of the ability of a feature $X(i)$ Discriminate class $+1$ from -1 is expressed as in Equation (6-6).

$$p(i) = \left| \frac{\mu_+(i) - \mu_-(i)}{\sigma_+(i) - \sigma_-(i)} \right| \quad (6-6)$$

where the capability of the associated feature for distinguishing the classes of the learning sample is measured by $d(i)$ or $p(i)$. Large values of $d(i)$ or $p(i)$ It means

the associated feature's expression level is highly differential between the two classes +1 from -1.

6.3.2 Sub-sampling Method

Assume we have a set of features $F = \{f_1, \dots, f_n\}$ and the feature is associated with an assigned significance value, i.e. $S = \{s_1, \dots, s_n\}$ where Equation (6-1) or Equation (6-6) is used to calculate the $s_i = d(f_i)$ or $p(f_i)$, the hybrid sub-sampling algorithm is in four stages to generate a subset of the feature ($\hat{F}$):

- i. Parameter calculation and feature filtering.
- ii. Feature ranking.
- iii. Feature partition.
- iv. Feature sub-sampling and recombination.

In the first stage, the features are filtered with their levels of significance ($d(f_i)$ or $p(f_i)$) calculates the parameters according to the number of sample features (n_F), the two parameters (α and β) and less than a given threshold (γ):

- i. The top group feature size (Denoted as F_{top}):

$$n_{top} = \alpha \times n_F \quad (6-7)$$

- ii. Number of features sampled from F_{top}

$$n_1 = \frac{n_{top}}{\beta} \quad (6-8)$$

- iii. The number of features selected from the group of the remaining features.

$$n_2 = n_F - n_1 \quad (6-9)$$

For instance, if we have $n_F = 25$, and the parameters $\beta = 4$ and $\alpha = 0.4$ then the size of F_{top} is $n_{top} = 10$, $n_1 = 2.5$ and $n_2 = 22.5$.

The features are ranked according to their significance and partitioned into two sets in the second stage. The top n_{top} features are contained in one set (Top significance features denoted as F_{top}) and the remaining features contained in the other set (denoted as F_{re}). The feature sub-sampling is performed using two approaches from the two feature sets. The top feature F_{top} , there is an equal probability of sampling because all the members have good discrimination power, while for the features in F_{re} , there are different probabilities among each of the

samples. The sampling is performed using random sampling. For the F_{re} In sampling, the significance of the features determines the probabilities of the sampling, i.e., a higher significance (denoted by $d(f_i)$ or $p(f_i)$) the feature has a higher sampling probability. The features are sampled in the end, respectively, from F_{top} and F_{re} are combined to generate the subset of one sample. Thus, sub-sample are run k-times, and then a sub-set of features produced denoted by $\widehat{F}_k$.

$$\widehat{F}_k = F'_{top} - F'_{re} \quad (6-10)$$

The significance of the feature can be normalised with Equation (6-11).

$$\widehat{s}_i = \frac{s_i}{\sum_{F_{re}}^n s_i} \quad (6-11)$$

The cumulative significance of the feature distribution function can be calculated using Equation (6-12).

$$cd_i = \sum_{j=1}^i \widehat{s}_j \quad (6-12)$$

6.3.3 Support Vector Classifiers

Many researchers have applied support vector classifiers (SVC) for different classifications and predictions with promising results. SVC minimises the objective function (the mean absolute or mean square error) by finding the hyperplanes producing the largest decision function value separation between the instances at the borderline of the two classes. However, the other modelling methods minimise the objective function of whole training instances [39].

Given data with label $D = \{(x_j, y_j)_k\}$ where, $j = 1, 2, \dots, n$; $k = 1, 2, \dots, m$, the construction of a decision function $f(x): R^n \rightarrow R$ is the SVM-based flight delay classification target, such that for every x_j , yields $f(x_j) > 0$ for $y_j = +1$, and $f(x_j) < 0$ for $y_j = -1$. A linear decision function $f(x) = W^T x + b$ or a non-linear decision function $f(x) = W^T \phi(x) + b$, where the non-linear transform function is $\phi(x)$ is employed by the SVC-based classifier. The $f(x)$ is determined by minimising $J(w, \varepsilon) = \frac{1}{2} \|W\|^2 + C \sum_{j=1}^l \varepsilon_j$ subjected to $y_j(W^T x_j + b) \geq 1 - \varepsilon_j$ (linear) or $y_j(W^T \phi(x_j) + b) \geq 1 - \varepsilon_j$ (non-linear), where the regularisation parameter is represented by $C > 0$ and the slack parameters are $\varepsilon_j \geq 0$ ($j = 1, 2, 3, \dots, l$).

For minimising the $J(w, \varepsilon)$, the linear combination of $\phi(x_j)$ can be the expression of W such that $w = \sum_{j=1}^l a_j y_j \phi(x_j)$. When we substitute w into $f(x) = W^T \phi(x) + b$ will result into $f(x) = \sum_{j=1}^l a_j y_j K(x_j, x) + b$, where the kernel function is represented by $K(u, v) = \phi^T(u)\phi(v)$. Then $a_j \geq 0$ ($j = 1, 2, \dots, l$) can be obtained from minimising $W(a_j) = \sum_{j=1}^l a_j - \frac{1}{2} \sum_{j=1}^l \sum_{k=1}^l a_j a_k y_j y_k K(x_j x_k)$ with the condition $0 \leq a_j \leq C$, and $\sum_{j=1}^l a_j y_j = 0$. SVC utilises three major kernel functions for classification are Linear $K(u, v) = uv$, polynomial $K(u, v) = (uv + 1)^d$ and the Gaussian function $K(u, v) = \exp\left(-\frac{\|u-v\|^2}{2\rho^2}\right)$. Where u, v are the features and their respective labels for the function K ; d is the set of polynomial degrees, while ρ is the length scale.

Boosting and Bagging Ensemble Machine Learners

These ensemble learning methods used the instance re-sampling scenario. The boosting algorithm applied in this research is AdaBoost, designed by the authors in [39]. A minimised weighted error is invoked on each iteration to AdaBoost by the learning algorithm on the training set and generates the base classifier accordingly. The bagging algorithm on each learning algorithm with a training set consisting of m training examples drawn from the random selection of the original dataset with replacement [4, 32, 40, 41]. The training examples' weights are adjusted according to the classification weighted error of the developed classifier model. The misclassified training examples are assigned more weights, while the training examples that were correctly classified are assigned fewer weights. A classification margin is induced in bagging by the random sampling procedure.

Additionally, the diversity in the resulting classification margin of bagging is due to the difference in feature subspace performance, and an ensemble learner needs to be accurate. It studies the average bagging error and then converts it to an optimisation problem for feature weight determination. The weights are assigned to the subspaces using a randomised technique in classification construction. The final classifier is constructed by individual classifiers voting for Boosting and Bagging [9, 42, 43].

Prediction Based on Proposed Voting Ensemble Method

We employed the majority voting method for making the final decision based on base classifiers' outputs. Given c_c classifiers $C_i(x): R^n \rightarrow \{-1, +1\}$, a weight $w_{i \in [0, 1]}$ is assigned to each classifier to show its significance. Assuming a new instance x , for each of the class predicted $C_i(x) \in \{-1, +1\}$ for $i = 1 - c_c$, the majority vote method

generates the final classification. Equation (6-13) represents the ensemble method of majority voting.

$$\mathbf{C}_{\text{ens}} = \text{sign} \left(\sum_{i=1}^{C_c} w_i c_i (x) \right) \quad (6-13)$$

In ensemble methods, the diversity of the ensemble is crucial for the success of the ensemble classifiers, which is the disagreement of the base classifiers [37]. A set of base classifiers is ideally selected to maximise accuracy and diversity to help seek a robust ensemble classifier. The effectiveness of compact ensemble committee selection by feature selection method has been proven by much research [36, 37, 44]. The proposed method in this paper characterises the behaviour of the candidate base classifier before the appropriate base classifiers are selected to construct a high-performance classification committee. Behaviours candidate base classifiers are characterised.

- i. Behaviours candidate base classifiers are characterised.
- ii. Base classifiers are grouped in terms of their classification behaviour.
- iii. An appropriate subset of the base classifiers is selected to construct the classification committee.

The proposed method's detail is shown in Algorithm 6-1.

Algorithm 6-1: Pseudocode of proposed ensemble Machine Learning method

```

1: Inputs: Subsets of features generated by the hybrid sub-sampling
   algorithm
2:      $F$ , the number of subsets of features
3:      $C_c$ , the expected number of ensemble classifiers
4: Outputs: A classification committee

5: Begin
6:    $F$  subsets of features and a training dataset with number of instances
7:   Group candidate classifiers based on classification behaviours
8:   Rank the classifiers based on their performance rate
9:   Select classifier with a high accuracy rate from different groups and
   construct committee (9)
10: End

```

The behaviour of the base classifiers associate feature subset). Selecting the best classifiers from different committees is based on the base classifier with the best

accuracy from each committee to ensure individual committee members' accuracy. The performance is similar in selecting the diverse classifiers from the candidates.

Dataset Description and Pre-processing

A flight that arrives or departs 15 minutes or more is considered delayed. In comparison, punctual or on-time flights are the ones that arrive or depart within the scheduled time, according to IATA [45, 46, 47, 48]. In this study, we employed data from the US commercial flight operation downloaded from the United States Department of Transportation from October to December 2000, with 1,048,576 instances, including all the missing variables, NA and inconsistencies records [49]. The provided features include airport information, plans; landing city; aircraft information; schedule and actual landing information summing up to 21 features from the raw dataset after eliminating all records with too much noise and at least four missing values. Some categorical variables, such as flight date, carrier information and aircraft tail number, cannot be processed directly in the flight delay dataset. Variables such as integer and floating-point data can be calculated directly. We then divided the dataset into training and testing sets with 70% and 30% ratios to train the model and predict the flight delay using the proposed algorithms.

Feature Selection

The role of feature choice cannot be overemphasised because it plays a key role in the model's results. Different features are selected for suitable models according to their characteristics to achieve better results. The embedded feature selection method uses the Machine Learning model for feature selection. The feature selection process integrates the learner training process and uses the feature score obtained during the training process to select features automatically.

Because ensemble learners such as random forest can score features, it makes it possible to select parameters of more importance to the model based on the ranking generated from the model's training, even with an unknown threshold. Therefore, we used the random forest to evaluate and select each feature, and we finally obtained the following 15 features for training the model. Some of the features have a strong correlation with each other. They need to be removed before the evaluation, for example, the actual elapsed time of an aircraft and other timing features that indicate a delay of an airline. Because flight delay is subject to actual departure or arrival delay, the categorical variables 0 and 1 are used as objects of prediction, where 0 means on-

time flight performance and 1 means delayed flight. Table 6-1 contains the list and description of this specific type of employed features in this research.

Table 6-1 Dataset features and description

Features	Data Type	Description
Arr_Delay	int64	Difference between scheduled and actual arrival time
Dep_Delay	int64	Difference between scheduled and actual departure time
Arr_Time	float64	Arrival time
Dep_Time	float64	Departure time
Taxi_In	int64	The number for an aircraft taxi coming in
Taxi_Out	int64	The number for an aircraft taxi going out
Actual_Elapsed_Time	float64	Elapse time
CRS_Elapsed_Time	float64	Schedule elapsed time
CRS_Arr_Time	float64	Scheduled arrival time
CRS_Dep_Time	float64	Schedule departure time
Day_of_Month	int64	Day of the month
Flight_Num	object	A unique number of the flight
Air_Time	float64	Time of flight in minutes
Dep_Del_Label	int64	Category flight delay and on time
Distance	int64	Different airport distances (miles)

Model Performance Evaluation and Validation

The criteria for evaluating and validating every Machine Learning model are important for the result measurement. There are different focuses for different scenarios. Generally, all classifier's pros and cons can be evaluated using precision, recall, F1-score and accuracy based on some objective functions or loss and cost function for convergence. The proportion of all correctly classified or judged positive samples is called accuracy. The proportion of the positive samples is known as the recall ratio because the flight time probability of arrival flight is far more than the probability of the flight delay. In this research, we pay more attention to on-time, precision and accuracy. At the same time, model accuracy must be higher than the accuracy of the sample containing most samples so that the model's efficiency is better than the result. A classification reference that ensures prediction accuracy is higher than classified with the largest proportion of all the variables to be explained into a category. Table 6-2 shows the confusion matrix for a given prediction problem.

Table 6-2 The Confusion matrix

Actual Class	Predicted Class		
	Positive		Negative
	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

The selection of a decision threshold in a binary classification problem influences the classifier's quality. Determining the decision threshold through the ROC-AUC and FPR-FNR plots is common, respectively. The receiver operating characteristic (ROC) curve depicts the same signal stimulus susceptibility on each point. The negative-positive class rate is the horizontal axis representing the true negative sample division into positive samples. In contrast, the vertical axis is the accuracy rate, and the area formed by the ROC is the AUC and the x-axis. The probability of the incorrect classification of the positive or negative samples when taking different thresholds is given by the FPR-FNR.

According to the airline's on-time statistics data and unbalanced data characteristics of the samples for binary forecast requirements, AUC-ROC and F1-score were selected as the pros and cons of the model. The evaluation criteria for the model are explained as follows in Equations (6-14) to Equation (6-19):

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (6-14)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (6-15)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (6-16)$$

$$\text{FPR} = \frac{FP}{FP+TN} \quad (6-17)$$

$$\text{FNR} = \frac{FN}{TP+FN} \quad (6-18)$$

$$\text{F1} = \frac{2*Precision*Recall}{Precision + Recall} \quad (6-19)$$

The harmonic average of the precision and recall are the F1-score having a value ranging as $F1 \in [0,1]$, and the optimal is when the model is 1. We employed the k-Folds cross-validation [50, 52].

6.4 Results and Discussion

In this section, we discussed the results and findings of our experiment. We conducted our experiment on a Personal Computer (PC) with an Intel (R) Core (TM) i7-9700 CPU with a processor speed of 3.00GHz and 32GHz RAM. We used libraries such as TensorFlow Core-2.4.1, TensorFlow GPU-2.4.1, NumPy-1.19.1, pandas-0.25.3, sci-kit learn-0.23.2, Scipy-1.5.2, PySimpleGUI-4.29.0, seaborn 0.11.2 and Matplotlib-3.3.1.

Our experimental study aims to verify the effectiveness of applying ensemble Machine Learning on flight datasets by comparing the performance of the proposed ensemble method and other widely used ensemble learners (Bagging, Boosting, Simple Averaging etc.). The performance comparison has been performed based on common conditions of the experiment, with additional data pre-processing performed before constructing the base classifier committee. The results from the experiment have shown that in conventional Machine Learning techniques (boosting, bagging and single classifier), there is a significant and consistent improvement in accuracy, precision and recall produced by our proposed method under different evaluation benchmarks.

Furthermore, there are competitive chances between the obtained results from this study and the methods existing in the literature. In [42] and [12], a boosting ensemble Machine Learning method was proposed, and the best accuracy of 80.44% and 87.72% were obtained for the arrival flight on-time dataset, respectively. While in [3], a stacking ensemble Machine Learning for flight delay prediction was compared, and the best accuracy of 82.22% was obtained for departure and arrival flight delay predictions.

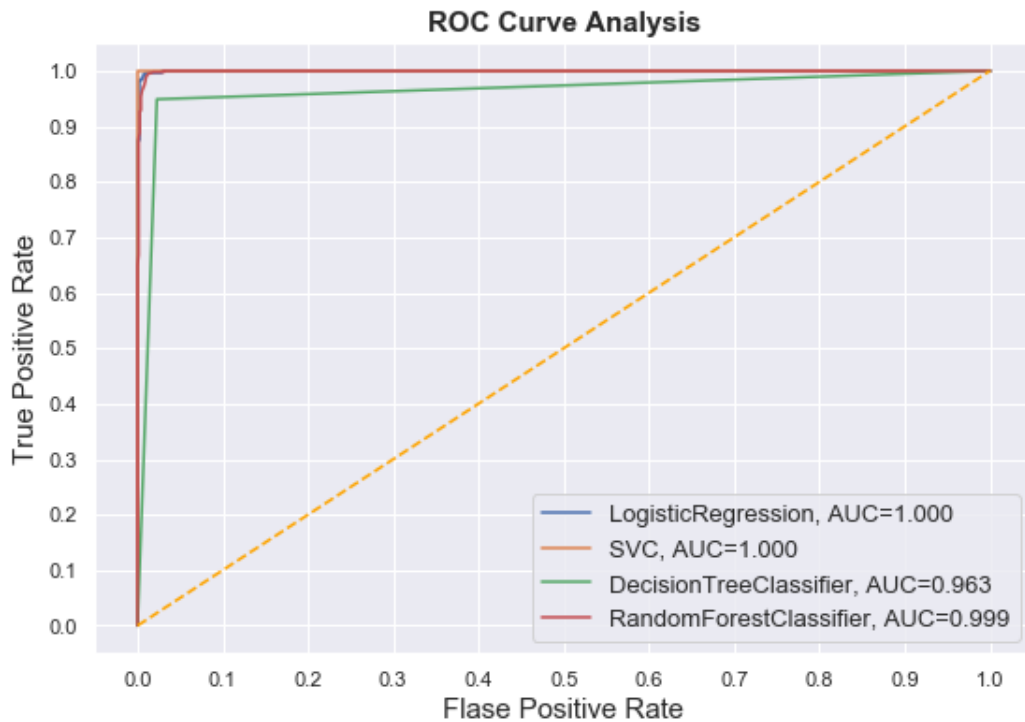


Figure 2Figure 6-2 ROC Curve of Models

Figure 6-2 shows the ROC (receiver operating characteristics) curve that measures the generalisation of the algorithm's ability. The AUC (under the curve) is the area under the ROC [3, 51]. An algorithm is considered better if it is closer to 1. We calculated and plotted each base committee algorithm's ROC and respective AUC scores, as shown in Figure 6-5. The Logistic Regression, Extra Gradient Boosting and Extra Trees reach 1.000, followed by the Random Forest with 0.999. The result of Bagging and Gradient Boosting was 0.997 and 0.996, respectively. The Decision Tree has 0.996, and it is the smallest compared to the base classifiers; this means that all the algorithms with lower values contribute less concerning the overall performance of the ensembles on the prediction of the arrival delay. Something worth noting is if there is an impact on the ensemble accuracy based on the committee algorithms' performance. As seen from the classifiers, the AUC are above 80%. It indicates the good overall sensitivity of the classifiers to predict flight delays. We assigned special names to all the ensembles, as shown in Table 6-3.

Table 6-3 Special names for ensembles

S/No	Ensemble Architecture	Special Name
1	Simple Averaging Ensemble Model	SAEM
2	Boosting Ensemble Model	BOEM
3	Stacking Ensemble Model	SEM
4	Hybrid Voting Ensemble Model	HVEM

The names proposed in Table 6-3 are not formal but a conventional approach for our convenience. It is to help us minimise long titles in our plots.

Table 6-4 Prediction results of different methods

S/No	Metrics	Methods			
		SAEM	BOEM	SEM	HVEM
1	Precision	0.78	0.81	0.93	0.97
2	Recall	0.67	0.80	0.94	0.98
3	Accuracy	0.67	0.80	0.95	0.99
4	F1-Score	0.58	0.78	0.90	0.97

Table 6-4 shows the ensemble algorithms comparison. We find that the HVEM model performs better with an accuracy of 0.99, precision of 0.97, recall of 0.98 and F1-score of 0.97, respectively, followed by SEM, having an accuracy of 0.95, recall of 0.94, and precision of 0.93 F1-score of 0.90, respectively. The BOEM is the next ensemble with a performance of 0.81 precision and 0.80 on recall and accuracy while having a 0.78 F1 -score. The SAEM performed poorly, with accuracy and recall of 0.67, a precision of 0.78 and an F1-score of 0.58. We run the experiment on all the models with the same number of base algorithms to help us understand the ensemble classifiers' impact on the prediction results. The high performance observed on the HVEM could be due to the ability of the committee base algorithms to effectively manage the trade between the stronger and the weaker learners in deciding the best results among the individual committee. The overall accuracy of the ensemble indicates great stability of the HVEM, as seen in Figure 6-3 with all other metrics plots. As mentioned, there is no "greatest algorithm" or a "multipurpose algorithm" in the Machine Learning field. Therefore, the HVEM algorithm could be a good solution for selecting algorithms, especially complex and enormous datasets like flight datasets.

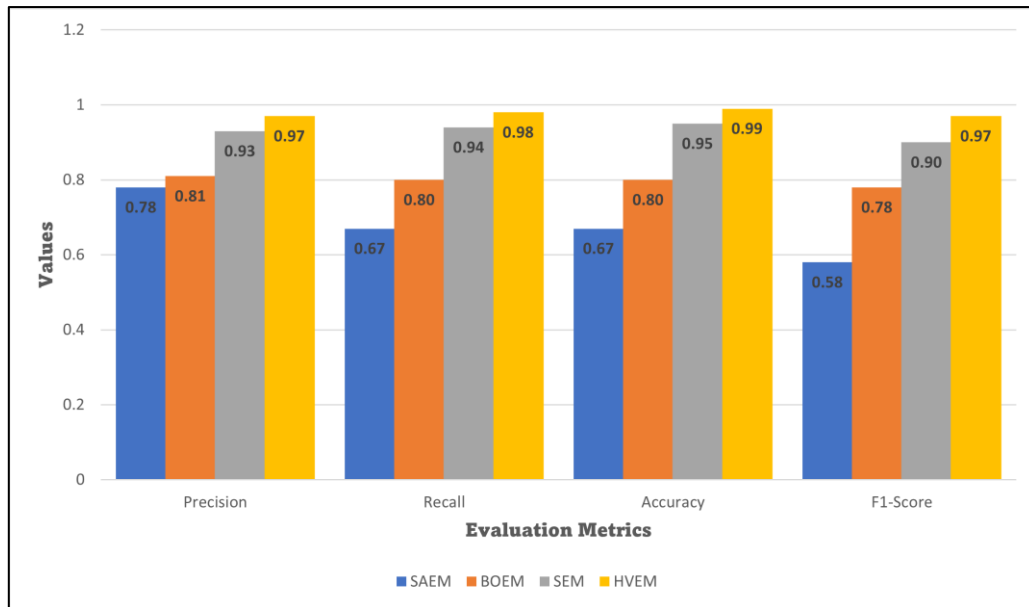


Figure 6-3 Prediction Results Plot of Different Methods

Confusion Matrix

We evaluated the classification efficacies of the ensemble classifiers with a confusion matrix, and the results are as follows in Table 6-5. The confusion matrix results of all the classifiers show the distribution of the actual versus predicted classes. The SAEM prediction result is the worst because of the high misclassification of the delayed class. For the True positive class, it is good with the model prediction of more than 746 flights correctly classified. Still, the model misclassified 395 flights, with only 55 flights' correct classification for the delayed class. In this classification, it can easily mislead decision-making. Looking at the BOEM model, the classification is good because the model correctly classifies 716 flights with no delays having only 34 misclassified and 241 delayed classes predicted. Although the BOEM model still has a high rate of misclassified delayed class 209, that is better than the SAEM since the correctly predicted for that class is higher than the misclassified flights. For the SEM, the model prediction is good for flights not being delayed and delayed. The model correctly classified the class not delayed with 750 flights as true positive and delayed with 448 flights as true negative class. The incorrect prediction is negligible since they are less than 1% of the class. Also, the HVEM model correctly predicted 750 flights not delayed and 426 flights delayed but incorrectly predicted 24 flights from the delayed class as not delayed. Our confusion matrix metrics aim to increase the true label class predictions for both the delay and not delayed classes. We can state that

the SEM model outperformed other models regarding the prediction results, followed by our proposed HVEM.

Knowing the importance of time management in the aviation sector. We can conclude on the variations and improvement from the different ensemble models by stating that it is very important for airlines and passengers to improve their flight knowledge for effective resource and time management, even if it is 0.0001%. It means that all the models implemented in this research, aside from the SAEM, can give more than 90% correct guidance to air passengers on their flight status and schedules. Fine-tuning the parameters of individual ensemble algorithms can lead to better prediction results.

Table 6-5 Confusion matrix results of the ensemble Machine Learning

		Predicted Class		
		No Delays	Delays	
SAEM	No Delays	746	4	
	Delays	395	55	
BOEM	No Delays	716	34	
	Delays	209	241	
SEM	No Delays	750	0	Actual Class
	Delays	2	448	
HVEM	No Delays	750	0	
	Delays	24	426	

Table 6-5 shows the results of the overall cross-validation of the bagging ensembles. We performed 50-Folds cross-validation across the classifiers committee base estimators to understand how well the classifiers understand new instances of the flight delay task. After 10-fold validation, the Decision tree and Random forest outperformed other models with an accuracy of 1.000. At the same time, Logistic regression had a value of 0.982, and Extra Tree had 0.968. Support Vector Classifier shows minimum accuracy of 0.683. This variation in the results is due to how each sub-sample is assigned to the fold. The cross validation pseudocode is as shown in Algorithm 2 [52].

Algorithm 6-2 K-fold cross validation pseudocode

```
1  Split TRAIN set into K chunks
2  For K = 1, 2, 3, ...K:
    (a) Set TRAINvalidation to be the kth chunk of dataset, and TRAINfit to be the
        other K-1 chunks
    (b) Fit each model to TRAINfit and evaluate how well it does on
        TRAINvalidation
    End for
3  Pick the model that has the best average TEST score
4  Retain that model on all of TRAIN, and output
```

Because the bagging ensemble performance was evaluated using the cross-validation algorithm, there is a need for statistical validation of the individual base estimators to measure the variation of each from the mean accuracy score. DecTree and RanFor are close to the mean accuracy score with a standard deviation (SD) of 0.000, while LogReg had SD of 0.021, ExTree with 0.022 and SupVecClas with 0.059. The SD value indicates how well each base estimator performed in the bagging ensemble.

Table 6-6 Results of bagging ensemble models with 10-fold cross-validation

Bagging Methods Base-Estimator	Special Name	Mean Accuracy Results	Standard Deviation
Logistic regression model	LogReg	0.982	0.021
Decision tree model	DecTree	1.000	0.000
Random forest model	RanFor	1.000	0.000
Support Vector Classifier model	SupVecClas	0.683	0.059

6.5 Conclusion

The major concern for the airports, airlines, passengers and other stakeholders is reducing or eliminating flight delays. However, minimising delays is not an easy task. Therefore, arrival delay prediction turns out useful. Several researchers have tried developing flight delay prediction models with good precision and accuracy. Our study proposed a hybrid ensemble learning approach based on supervised learning using voting aggregation. After performing the data stage pre-processing, the base classifiers were used to train the model. The experiment showed the highest accuracy

score of 0.99 for the proposed hybrid ensemble Machine Learning compared with other ensemble machine learners from the literature. The results from the similarity of our proposed method and the extra boost indicate how the voting method takes advantage of the mode value of the individual base classifier in deciding the model results.

Our proposed hybrid architecture can also be applied in predicting the departure or arrival of a train, bus or ship. Our model treats arrival flight delays but can also be applied to study departure delays. In this case, the transportation society can replace the airline or airport with the port or station and trip distance by the flight distance. The limitation of this research that can be subject to further research is that only the US flight information was applied to predict arrival flight delay. Another limitation is that we did not consider meteorological data.

The flight delay prediction using real-time flight datasets may be considered with other Machine Learning algorithms in the future. Moreover, weather influences flight delays and can be the focus. This current prediction model does not add exact weather-related features but does not make weather unimportant. Contrarily, it will be significant to study the influence of weather on flight delays. We will focus on establishing reasonable features to measure the high impact of weather on flight delays using Machine Learning to analyse the relationship between flight delay and weather. Furthermore, the Deep Learning model will be an interesting architecture to investigate. Finally, we would study and fulfilled all the needs and lacks in this research.

Acknowledgements and Funding

The Petroleum Trust Development Fund (PTDF) Nigeria partially funded this work through grant number PTDF/ED/OSS/PHD/DBB/1558/19 for the first author's PhD studies. We also thank the UKRI for the Covid-19 recovery grant under the budget code SA077N.

6.6 References

1. Bisandu DB., Moulitsas I., Filippone S. Social ski driver conditional autoregressive-based Deep Learning classifier for flight delay prediction. *Neural Computing and Applications*. Springer London; 2022; 34(11): 8777–8802. Available at: DOI:10.1007/s00521-022-06898-y
2. Bisandu DB., Homaid MS., Moulitsas I., Filippone S. A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective. *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK*. ACM; 2021. pp. 1–10. Available at: DOI:https://doi.org/10.1145/3505711.3505712
3. Yi J., Zhang H., Liu H., Zhong G., Li G. Flight Delay Classification Prediction Based on Stacking Algorithm. *Journal of Advanced Transportation*. 2021; 2021. Available at: DOI:10.1155/2021/4292778
4. Alharbi B., Prince M. A hybrid Artificial Intelligence approach to predict flight delay. *International Journal of Engineering Research and Technology*. 2020; 13(4): 814–822.
5. Perrell E. Embry-Riddle Aeronautical University. *AIAA SPACE 2013 Conference and Exposition*. 2013;
6. Kim YJ., Choi S., Briceno S., Mavris D. A Deep Learning approach to flight delay prediction. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. IEEE; 2016. pp. 1–6. Available at: DOI:10.1109/DASC.2016.7778092
7. M. Al-Tabbakh S., M. Mohamed H., H. El Z. Machine Learning techniques for analysis of Egyptian flight delay. *International Journal of Data Mining & Knowledge Management Process*. 2018; 8(3): 01–14. Available at: DOI:10.5121/ijdkp.2018.8301
8. Chen J., Li M. Chained predictions of flight delay using Machine Learning. *AIAA Scitech 2019 Forum*. 2019. pp. 1–25. Available at: DOI:10.2514/6.2019-1661
9. Alla H., Moumoun L., Balouki Y. A Multilayer Perceptron Neural Network with Selective-Data Training for Flight Arrival Delay Prediction. *Scientific Programming*. 2021; 2021. Available at: DOI:10.1155/2021/5558918
10. de Silva BM., Callaham J., Jonker J., Goebel N., Klemisch J., McDonald D., et al. Hybrid Learning Approach to Sensor Fault Detection with Flight Test Data. *AIAA Journal*. 2021; 59(9): 3490–3503. Available at: DOI:10.2514/1.j059943
11. Yan Z., Yang H., Li F., Lin Y. A Deep Learning approach for short-term airport traffic flow prediction. *Aerospace*. 2022; 9(1): 1–15. Available at: DOI:10.3390/aerospace9010011
12. Liu F., Sun J., Liu M., Yang J., Gui G. Generalized Flight Delay Prediction Method Using Gradient Boosting Decision Tree. *IEEE Vehicular Technology Conference*. 2020; 2020-May(May 2019). Available at: DOI:10.1109/VTC2020-Spring48590.2020.9129110
13. Sharma Priyadarshini M., Venturi S., Panesi M. Application of DeepOnet to model inelastic scattering probabilities in air mixtures. 2021; : 1–13. Available at: DOI:10.2514/6.2021-3144
14. Guo Z., Hao M., Yu B., Yao B. Detecting delay propagation in regional air

- transport systems using convergent cross mapping and complex network theory. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2022; 157(January 2021): 102585. Available at: DOI:10.1016/j.tre.2021.102585
15. Yu Y., Chen H., Yuan L., Zhang B. Flight delay classification warning based on evolutionary under-sampling bagging ensemble learning. 2021; 1205821(December 2021): 100. Available at: DOI:10.1117/12.2619725
 16. Sternberg A., Soares J., Carvalho D., Ogasawara E. A review on flight delay prediction. 2017; : 1–21. Available at: <http://arxiv.org/abs/1703.06118>
 17. Balakrishna P., Ganesan R., Sherry L. Accuracy of reinforcement learning algorithms for predicting aircraft taxi-out times: A case-study of Tampa Bay departures. *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd; 2010; 18(6): 950–962. Available at: DOI:10.1016/j.trc.2010.03.003
 18. Rodríguez-Sanz Á., Comendador FG., Valdés RA., Pérez-Castán J., Montes RB., Serrano SC. Assessment of airport arrival congestion and delay: Prediction and reliability. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November 2018): 255–283. Available at: DOI:10.1016/j.trc.2018.11.015
 19. Cheng C., Xu W., Wang J. A comparison of ensemble methods in financial market prediction. *Proceedings of the 2012 5th International Joint Conference on Computational Sciences and Optimisation, CSO 2012*. IEEE; 2012; : 755–759. Available at: DOI:10.1109/CSO.2012.171
 20. Henriques R., Feiteira I. Predictive modelling: Flight delays and associated factors, Hartsfield-Jackson Atlanta international airport. *Procedia Computer Science*. Elsevier B.V.; 2018; 138: 638–645. Available at: DOI:10.1016/j.procs.2018.10.085
 21. Choi S., Kim YJ., Briceno S., Mavris D. Cost-sensitive prediction of airline delays using Machine Learning. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. 2017; 2017-Sept. Available at: DOI:10.1109/DASC.2017.8102035
 22. Dou X. Flight Arrival Delay Prediction And Analysis Using Ensemble Learning. 2020; (It nec): 836–840. Available at: DOI:10.1109/itnec48623.2020.9084929
 23. Campanelli B., Fleurquin P., Arranz A., Etxebarria I., Ciruelos C., Eguíluz VM., et al. Comparing the modelling of delay propagation in the US and European air traffic networks. *Journal of Air Transport Management*. 2016; 56(Part A): 12–18. Available at: DOI:10.1016/j.jairtraman.2016.03.017
 24. Hendrickx R., Zoutendijk M., Mitici M., Schafer J. Considering Airport Planners' Preferences and Imbalanced Datasets when Predicting Flight Delays and Cancellations. *IEEE*; 2021; : 1–10. Available at: DOI:10.1109/dasc52595.2021.9594367
 25. Mueller ER., Chatterji GB. Analysis of aircraft arrival and departure delay characteristics. *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*. 2002. pp. 1–14. Available at: DOI:10.2514/6.2002-5866
 26. Wu Q., Hu M., Ma X., Wang Y., Cong W., Delahaye D. Modelling Flight Delay Propagation in Airport and Airspace Network. *IEEE Conference on*

- Intelligent Transportation Systems, Proceedings, ITSC. 2018; 2018-Novem: 3556–3561. Available at: DOI:10.1109/ITSC.2018.8569657
27. Tu Y., Ball MO., Jank WS. Estimating flight departure delay distributions - A statistical approach with long-term trend and short-term pattern. *Journal of the American Statistical Association*. 2008; 103(481): 112–125. Available at: DOI:10.1198/016214507000000257
 28. Wu F., Wu L. DeepETA: A Spatial-Temporal Sequential Neural Network Model for Estimating Time of Arrival in Package Delivery System. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2019; 33: 774–781. Available at: DOI:10.1609/aaai.v33i01.3301774
 29. Abdella JA., Zaki N., Shuaib K., Khan F. Airline ticket price and demand prediction: A survey. *Journal of King Saud University - Computer and Information Sciences*. The Authors; 2019; (xxxx). Available at: DOI:10.1016/j.jksuci.2019.02.001
 30. Pyrgiotis N., Malone KM., Odoni A. Modelling delay propagation within an airport network. *Transportation Research Part C: Emerging Technologies*. 2013; 27: 60–75. Available at: DOI:10.1016/j.trc.2011.05.017
 31. Xu N., Sherry L., Laskey KB. Multifactor model for predicting delays at U.S. airports. *Transportation Research Record*. 2008; (2052): 62–71. Available at: DOI:10.3141/2052-08
 32. Belcastro L., Marozzo F., Talia D., Trunfio P. Using scalable data mining for predicting flight delays. *ACM Transactions on Intelligent Systems and Technology*. 2016; 8(1). Available at: DOI:10.1145/2888402
 33. Horiguchi Y., Baba Y., Kashima H., Suzuki M., Kayahara H., Maeno J. Predicting Fuel Consumption and Flight Delays for Low-Cost Airlines. *Innovative Applications of Artificial Intelligence (IAAI) Conference*. 2017; : 4686–4693.
 34. Moretti F., Pizzuti S., Panzieri S., Annunziato M. Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modelling. *Neurocomputing*. Elsevier; 2015; 167: 3–7. Available at: DOI:10.1016/j.neucom.2014.08.100
 35. Sataloff RT., Johns MM., Kost KM. *Ensembles in Machine Learning Applications*.
 36. Mohareb F., Papadopoulou O., Panagou E., Nychas GJ., Bessant C. Ensemble-based support vector machine classifiers as an efficient tool for quality assessment of beef fillets from electronic nose data. *Analytical Methods*. Royal Society of Chemistry; 2016; 8(18): 3711–3721. Available at: DOI:10.1039/c6ay00147e
 37. Peng Y. A novel ensemble Machine Learning for robust microarray data classification. *Computers in Biology and Medicine*. 2006; 36(6): 553–573. Available at: DOI:10.1016/j.combiomed.2005.04.001
 38. Cavalcanti PG., Scharcanski J., Baranoski GVG. A two-stage approach for discriminating melanocytic skin lesions using standard cameras. *Expert Systems with Applications*. 2013; 40(10): 4054–4064. Available at: DOI:10.1016/j.eswa.2013.01.002
 39. Levi Y., Bekhor S., Rosenfeld Y. A multi-objective optimisation model for urban planning: The case of a very large floating structure. *Transportation Research Part C: Emerging Technologies*. Elsevier; 2019; 98(November

- 2018): 85–100. Available at: DOI:10.1016/j.trc.2018.11.013
40. Yu B., Guo Z., Asian S., Wang H., Chen G. Flight delay prediction for commercial air transport: A Deep Learning approach. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier; 2019; 125(March): 203–221. Available at: DOI:10.1016/j.tre.2019.03.013
 41. Fernandes N., Moro S., Costa CJ., Aparício M. Factors influencing charter flight departure delay. *Research in Transportation Business and Management*. Elsevier; 2019; 34(December): 100413. Available at: DOI:10.1016/j.rtbm.2019.100413
 42. Yao S. Flight Arrival Delay Prediction Using Gradient Boosting Classifier. *Emerging Technologies in Data Mining and Information Security*. Yao, S. (2019). *Emerging Technologies in Data Mining and Information Security*. In *Emerging Technologies in Data Mining and Information Security* (Vol. 813). <https://doi.org/10.1007/978-981-13-1498-8Secur>. Springer Singapore; 2019. 871–877 p. Available at: DOI:10.1007/978-981-13-1498-8
 43. Mangorrey E., Puranik TG., Pinon-Fischer OJ., Mavris DN. Classification, Analysis, and Prediction of the Daily Operations of Airports Using Machine Learning. 2020; (January): 1–18. Available at: DOI:10.2514/6.2020-1196
 44. Hesam M., Sultana N., Millward H., Liu L. Ensemble learning activity scheduler for activity based travel demand models. *Transportation Research Part C*. Elsevier Ltd; 2021; 123(February): 102972. Available at: DOI:10.1016/j.trc.2021.102972
 45. de Juniac A. International Air Transport Association (IATA) - Annual review 2019. 75th Annual General Meeting, Seoul, June 2019. 2019; Available at: <https://www.iata.org/publications/Documents/iata-annual-review-2019.pdf> (Accessed: 3 May 2021)
 46. IATA. AI in Aviation. 2018; (June): 1–20. Available at: <http://www.iata.org/publications/Documents/AI-White-Paper.pdf>
 47. Industry High-level Group. Aviation Benefits Report. 2019; : 76.
 48. Boksberger P. Aviation Systems. *Aviation Systems*. 2011; : 157–170. Available at: DOI:10.1007/978-3-642-20080-9
 49. Fleurquin P., Ramasco JJ., Eguiluz VM. Systemic delay propagation in the US airport network. *Scientific Reports*. 2013; 3. Available at: DOI:10.1038/srep01159
 50. Moreno-Torres JG., Saez JA., Herrera F. Study on the impact of partition-induced dataset shift on k-fold cross-validation. *IEEE Transactions on Neural Networks and Learning Systems*. 2012; 23(8): 1304–1312. Available at: DOI:10.1109/TNNLS.2012.2199516
 51. Chander S., Padmanabha V., Mani J. Jaya Spider Monkey Optimisation-driven Deep Convolutional LSTM for the prediction of COVID'19. *Bio-Algorithms and Med-Systems*. 2020. Available at: DOI:10.1515/bams-2020-0030
 52. Domke IJ. Overfitting , Model Selection , Cross Validation , Bias-Variance. *Machine Learning*. : 1–9.

Chapter 7: A Deep Bi-LSTM Machine Learning Method for Flight Delay Prediction Classification

This paper proposes a classification approach for flight delays using Bidirectional Long Short-Term Memory (BiLSTM) and Long Short-Term Memory (LSTM) models. Flight delays are a major issue in the airline industry, causing inconvenience to passengers and financial losses to airlines. The BiLSTM and LSTM models, powerful Deep Learning techniques, have shown promising results in a classification task. In this study, we collected a dataset from the United States (US) Bureau of Transportation Statistics (BTS) of flight on-time performance information and used it to train and test the BiLSTM and LSTM models. We set three criteria for selecting highly important features to train and test the models. The performance evaluation of the models and Confusion matrix shows that BiLSTM outperforms the LSTM model. In evaluating the models using the Mathews Correlation Coefficient (MCC), the BiLSTM model offers a better correlation of 0.99 between the original and predicted classes. Our experiment shows that for predicting flight delays, the BiLSTM model takes advantage of the forward and backward hidden sequences and the deep neural network for performance exploration and exploitation to achieve high accuracy, recall, and F1-Score. Our findings suggest that the BiLSTM model can effectively predict flight delays and provide valuable information for airlines, passengers and airport managers.

7.1 Introduction

Recently, flight delays have increased due to the rapid growth of air transportation system demand and the world's globalisation influence in the 21st century [1, 2]. The International Civil Aviation Organisation (ICAO) report in 2019 shows that there has been a clear sign of constant growth in the aviation industry. There were 155 million commercial passengers between 2018 and 2019, resulting from an expansion year-on-year with a 6.07% increase [1, 3]. This growth in demand for air travel outpaces airport capacity expansions. Also, the Civil Aviation Administration (CAA) of China reported in 2020 that there was an 88.52% increase of the on-time rate of flights from civil aviation [4] and an 80.13% significant increase in 2018 [5]. However, civil aviation still faces a major problem

of flight delays [6–8]. The negative effect caused by flight delays on all the stakeholders cannot be overemphasised [9]. It leads to losses of a huge amount of money [10, 11], also making both airlines and passengers have conflicts [12], and operational efficiency is being reduced [13] in the entire civil aviation sector [14]. Flight delay characteristics research and prediction model establishment have improved the services of the decision support department of the aviation control and airlines [15, 16], enhanced operational efficiency of civil aviation [17, 18] and reduced loss [19].

Air transportation delay cost in 2007 was estimated to total \$33 billion in the United States (US), and \$16.7 billion of the delays was associated with passengers [19, 20]. Hence, this indicates how important the issue of flight delay is in the air transportation system, and management needs to pay close attention to understand and evaluate its occurrences effectively. The chain reaction of other businesses, such as the non-aviation-related ones, is also affected by the impact of flight delays, among other financial and operational aspects of civil aviation. The overall goal is to minimise unnecessary cost but enhancing the performance of the flight delay prediction model is desired to help avoid and mitigate risk for the stakeholders in the commercial aviation industry. There are internal and external factors that have been identified to cause flight delays. The controllable factors by the airline are referred to as the internal factors, such as the availability of the gate, while the external factors depend on the uncontrollable factors, such as baggage handling, passenger handling, and bad weather. The complexity of understanding datasets related to flight delay prediction makes it one of the aviation industry's most challenging problems. Also, internal and external factors significantly increase the problem's complexity due to unexpected events.

One of the most important performance indicators to effectively assess the service quality of airline and airport management is the punctuality of schedule flights. In theory, the time required to design practices may not be possible because of uncontrollable factors such as sudden pilot sickness and bad weather. Reports from Federal Aviation Administrator (FAA) and standards for operation show that any flight that landed 15 minutes after the originally scheduled time is

considered “*delayed*”. There are many regulations by many airports requiring airlines to compensate passengers who experience delays exceeding a certain threshold. It is common to have congestion, which makes the delays common too. Delay can lead to detrimental effects due to its consecutive propagation. Hence, it will be more beneficial to all involved stakeholders in the air transportation system to predict the occurrence as it will help them be prepared and make all the necessary responses to avoid further consequences.

The advances in data analytics and Artificial Intelligence have motivated research on the possible application of flight delay in predicting the use of commercial airlines. Several approaches are applied in the analysis and prediction of flight delays. These are Machine Learning, statistical analysis and Deep Learning [3]. Therefore, understanding and critically evaluating the different techniques align with the application domain. Neural networks have been proven to perform better in state-of-the-art Deep Learning methods when applied to complex datasets with higher accuracy. However, the real-time video and images dataset was applied to feed the models with a complex structure considered unstructured or semi-structured datasets. But it is highly significant that the viability of applying a bidirectional long short-term memory (LSTM) type of recurrent neural network (RNN) model with structure and offline dataset such as for flight delays to study the performance comparison similarity with others. Also, based on our findings, there are limited or few contributions in the research area of flight delay predictive modelling in which the bidirectional LSTM model is used to predict flight delays.

In this paper, we utilise a bidirectional deep LSTM architecture to perform flight delay analysis and prediction using flight on-time datasets obtained from the US Department of Transportation, Bureau of Statistics (BTS). The primary objective is to investigate bidirectional deep LSTM architecture with the deep neural network and unidirectional LSTM architecture in the flight delay predictive task evaluated with well-known benchmark metrics.

The rest of the paper is as follows. Section 7.2 presents other past studies of flight delay analysis and predictive tasks. Section 7.3 explains the material and methods of this study with the proposed approach. The results and discussion

are detailed in Section 7.4, while Section 7.5 concludes the studies with future directions.

7.2 Related Work

Researchers and industry practitioners have faced challenges in accurately predicting flight delays for decades. However, the applicability of recent studies using state-of-the-art approaches such as Machine Learning, data, and Deep Learning has demonstrated better prediction results on flight delays than statistical approaches [21–22]. As a result, several research studies have been conducted to assess the factors and impact of non-meteorological and meteorological conditions on flight delays.

Bisandu et al. in [3] utilise a special type of deep recurrent neural network (RNN) known as deep long short-term memory (LSTM) and social ski driver conditional autoregressive-based Deep Learning to study non-weather impacted delays using datasets from the US Bureau of Statistics. Their experiment shows the importance of data pre-processing and fine-tuning Deep Learning algorithms with optimisation algorithms to improve the model's accuracy, error rate, and reduced computational requirements compared to other metaheuristic methods. In [8] and [23], the authors discuss the relevance of data in predicting flight delays and identify major methods such as Machine Learning, Deep Learning and statistical methods as the currently applied methods in researching flight delay predictive tasks. Kim et al. [21] proposed different architectural designs and implementations of LSTM and RNN in predicting flight delays using sequences of thresholds. In [22], the authors used convolutional LSTM (Conv-LSTM) to predict airport flight delays using temporal and spatial characteristics in China civil aviation.

In [23], the author explores the characteristics of an aircraft's arrival and departure delay to develop a time series model that determines the probability of the correlation of the features using poison and normal probability distribution with density function. The authors in [24] combine a mathematical weight value with linear and non-linear kernels of regression algorithms to propose a model for predicting flight delays using data from a Swedish airline softwarelinx. Also, authors in [16] collected data using surveillance-broadcast aviation platforms and

used random forests based on LSTM to explore factors influencing flight delays. Balcastro et al. [25] incorporate parallel algorithms using MapReduce with weather datasets to predict arrival flight delays. Finally, in [26], Chen and Li propose a multi-label random forest propagation and classification model using an optimal feature selection process that predicts chains of delays considering the initial departure.

In [27], the authors proposed a method for predicting the flight departure delay at Nanjing Lukou International Airport. They apply four different supervised Machine Learning algorithms known as support vector machine, Light Gradient Boosting Machine (LightGBM), multiple linear regression and extremely randomised trees using a comparative study to explain their suitability based on their experiments. Bisandu et al. [28] compare deep feedforward neural network architecture with shallow architectures using a filter-based feature selection technique on non-weather impacted flight delays. The experiment results show how important parameter fine-tuning makes a model more reliable in predicting departure and arrival flight delays. Manna et al. [29] analyse air traffic data using a gradient boost decision tree, and the model produces higher accuracy based on their experiment. In [30], Chakrabarty used a grid search hyper-parameter tuning and gradient boosting classifier model for analysing and predicting the arrival delay of American Airlines using the top must include busiest airports with a binary classification technique. Authors in [31] employed a single-layer neural network with logistic regression and a decision tree to detect whether an arrival flight will be delayed using only the top three features from the feature importance results as the model inputs. In [32] predicted arrival delays at Tokyo Airport using queue analysis artificial neural network (ANN) with Rectifier Linear Unit (ReLU).

In [33], Gopalakrishnan and Balakrishnan proposed a flight delay prediction model based on origin-destination pairs using a two-hour time horizon with an architecture of multiple artificial neural networks and the Markov Jump Linear System. The authors in [34] proposed a model for flight delay with novel factors known as crowdedness in the Beijing International Airport airport route using a deep belief network and support vector regression (DBN_SVR) on the top layer of the proposed model. In [35] offered a new method to predict incoming flights

at John F. Kennedy (JFK) International airport using ANN and Defect of Modules Prediction (DPM). The authors in [36] applied stacked autoencoders (SAE) for traffic flow prediction using a greedy Deep Learning architecture, which learns best from the features. In [37] author proposed an airport delay prediction model using BiLSTM sequence learning with spatiotemporal analysis. Their model shows better stability and accuracy than the other methods in their experiment. Karim et al. [38] developed a model for time series classification using fully connected convolutional networks and LSTM with a novel fine-tuning technique for the LSTM cell, despite the remarkable results from the previous studies. there is a common conclusion among them, and it is the fact that there is a need to perform analysis and propose additional models based on new contexts and perspectives. Also, to check the suitability and viability of the prediction models considering the change in every aspect of the transportation industry, especially the air transport sector.

With the advent of neural networks and Deep Learning with LSTM architectures, predictive model performance has shifted, especially in flight delay tasks. Previous revised studies have used complex data such as images and videos with models [2, 29]. However, based on our knowledge, we are among the first to compare bidirectional deep LSTM and one-directional LSTM on flight on-time datasets. Therefore, this paper investigates the efficacy and viability of bidirectional deep LSTM on flight on-time records to perform flight delay predictive tasks. This study intends to contribute to the air transportation system by proposing an innovative, accurate flight delay prediction model using bidirectional deep LSTM architecture and the US BTS dataset.

7.3 Materials and Methods

This study employed an airline performance on-time dataset downloaded from the US BTS [39]. Specifically, we have used the October to December 2013 dataset for the experiment. There were 29 features initially in the dataset before pre-processing.

7.3.1 Data Description and Pre-processing

The information in the dataset is from October to December flight on-time US BTS Table 7-1 shows the 29 features with their corresponding definitions [40] and the attribute types for each feature [41]. They needed to be reduced further because of constraints in computation, inconsistencies and empty cells or non-required features. We split the training and testing datasets into 70% and 30%. We grouped the delay time into two for the presence of delay and no delay blocks instead of applying the exact delay time. The delay block and flight that arrived or departed 15 minutes later is considered *delayed*, known as Class 1, while the non-delay *on-time* block is any flight that arrived and departed in less than 15 minutes, called Class 0. From the 29 features, we selected six features for training the model, which are the most relevant features to the flight delay predictive task. Table 7-1 Dataset features and their descriptions

S/No	Feature	Attribute Type	Details
1	Year	Numerical (Discrete)	Example, 2000
2	Months	Numerical (Discrete)	Example, 12
3	DayOfMonth	Numerical (Discrete)	Example, 01-31
4	DayOfWeek	Numerical (Discrete)	Example, 1 (Monday) - 7 (Sunday)
5	DepartureTime	Numerical (Continuous)	Example, 1456
6	ScheduledDepartureTime	Numerical (Continuous)	Example, 1456
7	ArrivalTime	Numerical (Continuous)	Example, 1456
8	ScheduledArrivalTime	Numerical (Continuous)	Example, 1456
9	UniqueCarrierCode	Categorical (Nominal)	Example, PS
10	FlightNumber	Numerical (Discrete)	Example, 1454
11	PlaneTailNumber	Categorical (Nominal)	Example, N923XJ
12	ActualElapsedTime	Numerical (Continuous)	Example, 194
13	ScheduledElapsedTime	Numerical (Continuous)	Example, 200
14	FlightTime	Numerical (Continuous)	Example, 94
15	ArrivalTime	Numerical (Continuous)	Example, 1015
16	DepartureTime	Numerical (Continuous)	Example, 1015
17	Origin	Categorical (Nominal)	Example, LHR
18	Destination	Categorical (Nominal)	Example, MAN
19	Distance	Numerical (Continuous)	Example, 1448
20	TaxiIn	Numerical (Continuous)	Example, 10
21	TaxiOut	Numerical (Continuous)	Example, 10
22	Cancelled	Binary (Categorical)	Example, 0 or 1
23	ArrDel	Numerical (Nominal)	Example, -11, 12, 0
24	DepDel	Numerical (Nominal)	Example, -10, 13, 0
25	CarrierDelay	Binary (Categorical)	Example, 0 or 1
26	WeatherDelay	Binary (Categorical)	Example, 0 or 1

27	AviationSystemDelay	Binary (Categorical)	Example, 0 or 1
28	SecurityDelay	Binary (Categorical)	Example, 0 or 1
29	LateAircraftDelay	Binary (Categorical)	Example, 0 or 1

From Table 7-1, initially, we removed some features that do not add much to our model performance. These features were removed because some had information after the delay had been confirmed, complete zero in all the cells, or had the same value that did not impact the model's training performance. Our conditions and criteria for the feature selection process are: (1) We focus on delays arising in the departure airport. Therefore, features associated with post-departure information are not considered. For example, *TaxiIn* is omitted due to information about when the flight wheels arrive at the destination gate runway; others are (*ArrTime*, *AirTime*, *ActualElapsedTime* and *ScheduledElapsedTime*. In addition, there are 5 other variables (*NASDelay*, *WeatherDelay*, *SecurityDelay*, *LateAircraftDelay* and *CarrierDelay*). (2) Features having no information about flight delays, such as *TailNum*, *Year*, *UniqueCarrier*, *FlightNum*, *DestCode*, *OriginCode*, *Diverted*, *CancellationCode*, *DayofWeek*, *DayofMonth*, and *Cancelled*. (3) We have calculated the correlation coefficient to estimate how a pair of variables influences one another and to detect collinearity between variables. Variables such as *Delay_Level*, *DepTime*, and *ScheduledArrivalTime* are omitted due to high correlation with *Month*, *DepDelay* and *ArrDelay*. We used the following six features in training our model: *Months*, *ScheduleDepTime*, *Distance*, *TaxiOut*, *DepDelay*, and *ArrvDelay*. Table 7-2 shows the features we utilised in training our model. The remaining feature instance with incomplete information, such as NA, were also removed to process the dataset further to be suitable for the proposed model.

Flight delay is a complex phenomenon affected by numerous factors, including carrier, weather conditions, air traffic congestion, security, and mechanical issues. However, some of the essential features that affect flight delays are: The month of the year can significantly impact flight delays, as different months may experience different weather patterns or holiday travel patterns that can impact flight schedules. For instance, holiday seasons may experience more delays due to people travelling more spent time with their loved

ones. Distance: The distance of the flight can also impact its delay, as longer flights may encounter more air traffic congestion, require more time for boarding and unloading, and may have more complicated routing. ScheduleDepTime: The scheduled departure time can affect the delay of a flight as certain times of day, such as peak travel times, may experience more congestion on the runway or in the air. TaxiOut: The time it takes for an aircraft to taxi from the gate to the runway, known as taxi-out time, can significantly impact the delay of a flight. Longer taxi-out times can lead to a delayed departure, causing the flight to arrive late. DepDelay: The departure delay, which measures the time difference between the scheduled and actual departure times, is one of the most crucial features affecting flight delay. A longer departure delay may lead to more significant delays throughout the flight, impacting the arrival time. ArrDelay: Finally, the arrival delay, which measures the time difference between the scheduled arrival time and the actual arrival time, is also a critical factor in a flight delay. Longer arrival delays can disrupt passengers' schedules, cause missed connections, and impact airline performance.

Table 7-2 Features used in training our model

S/No	Feature	Type	Description
1	Months	Numerical(Discrete)	Recorded flight months from October to December.
2	Distance	Numerical (Continuous)	Origin and destination distance in miles
3	ScheduleDepTime	Numerical (Continuous)	Departure schedule.
4	TaxiOut	Numerical (Continuous)	Taxi-out time in minutes.
5	DepDelay	Numerical (Nominal)	Time difference between a scheduled and actual departure.
6	ArrDelay	Numerical (Nominal)	Time difference between a scheduled and actual arrival.

The dataset contains ten airports with their codes, as shown in Table 7-3.

Table 7-3 Airports codes

S/No	Airport Codes	Name of Airport
1	ATL	Atlanta
2	ORD	Chicago
3	LAX	Los Angeles
4	SAN	San Diego
5	MSP	Minneapolis
6	PHX	Phoenix
7	PHL	Philadelphia
8	SFO	San Francisco
9	DEN	Denver
10	BOS	Logan

We used min-max normalisation for our data because it is all in numerical form. We transformed the data to fit the training and testing set. This process is important because all features may have various data types. The numerical differences are eliminated because of the different range of values when computing. A value x is converted into x' range $[max_new - min_new]$ as in Equation (7-1):

$$x' = \frac{x - x_{min}}{x - x_{max}} \times [max_new - min_new] + min_new \quad (7-1)$$

where the range of the transform values is denoted from min_new to max_new . We set $min_new = 0$ and $max_new = 1$. The transform values were then used as input into the BiLSTM architecture.

7.3.2 Proposed methodology

The methodology proposed for this study is described in detail in this section, concentrating on all the aspects that assist improve the tuning capabilities of the methods we are compared. Algorithm 7-1 shows the clear steps taken in the proposed methodology. The proposed BiLSTM model pseudocode is presented in Algorithm 7-1.

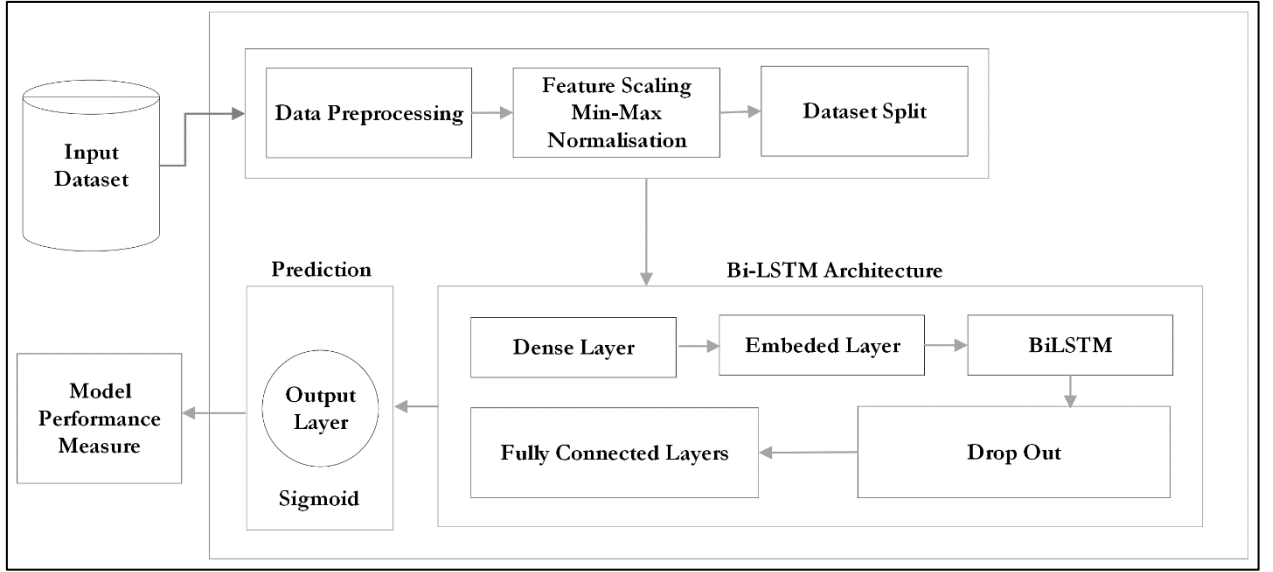


Figure 7-1 Proposed Prediction Methodology

Batch Normalisation Layer

The training dataset is acquired batch-by-batch. The distributions are unstable and non-uniform, and the network parameters in each training cycle must be fitted, showing the model's significant convergence. Batch normalisation is used for reparameterization. The mean α_D and variance α_D^2 is determined by the batch normalisation technique for each training dataset batch and the scale adjusted for the original dataset to unity-variance and zero-mean. The shifted data $\hat{a}_1$ are applied to weight and bias to help enhance expressive power. Equation (7-2) to Equation (7-5) shows the computation of the batch normalisation. The batch normalisation algorithm does the reparameterization update coordination across the neural network layers.

$$\alpha_D = \frac{1}{n} \sum_{i=1}^n a_i \quad (7-2)$$

$$\alpha_D^2 = \frac{1}{n} \sum_{i=1}^n (a_i - \alpha_D)^2 \quad (7-3)$$

$$\hat{a}_1 = \frac{a_i - \alpha_D}{\sqrt{\alpha_D^2 + \epsilon}} \quad (7-4)$$

$$b_i = \gamma \hat{a}_1 + \beta \quad (7-5)$$

where b_i , ε , γ and β is the final normalised value and parameters that make sure the batch normalisation learns the identity function in a few cases.

Bidirectional Long Short-Term Memory (LSTM) Architecture

The most sophisticated type of Machine Learning available today is Deep Learning. It has brought about the growth in various models with neural networks for application in solving real-world problems. We utilised an effective Deep Learning method known as long short-term memory because of its memory-oriented features to model and analyse flight delays. BiLSTM, as a Deep Learning approach, is efficient in analysing and extracting important data features needed for a predictive task. It is an extension of the Recurrent Neural Network (RNN). The LSTM structure [42] was designed to address the *vanishing gradient* problem of the RNN structure. The LSTM Cell structure contains an input gate, output gate, forget gate and memory unit [43]. The memory block structure of the one-layer neural network controls the forget gate. Equation (7-6) can determine the gate's activation.

$$f_a = \sigma(W[x_a, h_{a-1}, C_{a-1}] + b_v) \quad (7-6)$$

where x_a denotes the sequence of inputs; h_{a-1} is the output of the previous block; C_{a-1} denotes the block memory of the previous LSTM; and the bias vector is denoted by b_v . The individual weight vectors for each input are denoted by W , while the logistic sigmoid activation function is represented by σ .

The basic NN with the *tanh* activation function has an input gate as part of its structure with new memory created by the block effect of the prior memory. The operations can be computed using Equation (7-7) and Equation (7-8).

$$i_a = \sigma(W[x_a, h_{a-1}, C_{a-1}] + b_i) \quad (7-7)$$

$$C_a = f_a \cdot C_{a-1} + i_a \cdot \tanh([x_a, h_{a-1}, C_{a-1}] + b_c) \quad (7-8)$$

LSTM with one-way movement relies on past data, but it is always inadequate. Dependencies are prevented by consciously remembering long-term information, which in practice is the behaviour of the LSTM by default. The BiLSTM only analyses data in two directions. Two values are held by the hidden layer of the

BiLSTM [44], which are employed for forward computation and reverse calculation. The final output prediction performance of the BiLSTM is enhanced by the two values which determine the output [37]. Figure 7-2 shows the architecture of a BiLSTM.

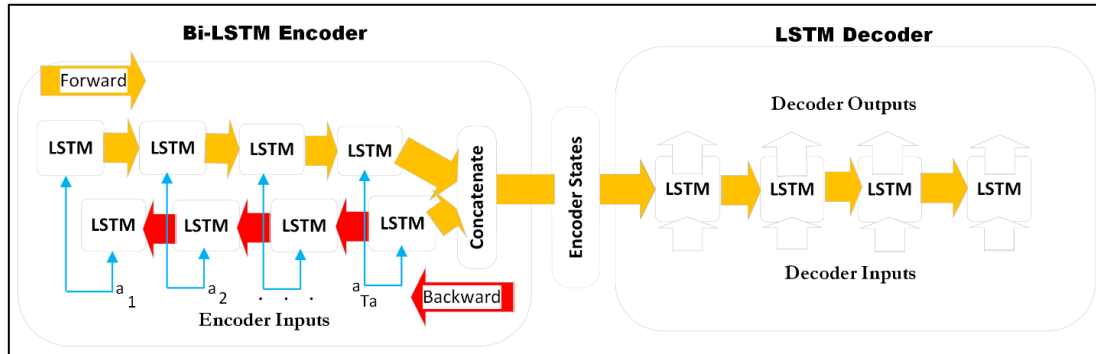


Figure 7-2 An Architecture of a BiLSTM

Figure 7-2 shows two recurrent components, a forward and a backward recurrent component. The forward component computes the hidden and cell states, similar to a standard LSTM, while the backward component computes by a reverse-chronological input sequence. For example, it is taken from Ta to 1 timestamp. The backward component usage creates a way in which future data are captured by the network and tries to learn its weights, respectively. It helps capture some dependencies by the network that a standard unidirectional LSTM might not have been able to capture. BiLSTM is also good in Natural Language Processing because of its ability to capture input sequence dependencies quite well. The forward components' hidden, and cell states differ from the backward components. Therefore, the hidden and cell states of the forward component are concatenated with the backward components to get encoded.

Unidirectional Long Short-Term Memory Architecture

The sequence of time series is taken as input by the LSTM encoder (each LSTM cell takes a one-time step), creating an input sequence for encoding. A vector consisting of hidden and cell states is created by encoding all the LSTM cells. The LSTM decoder receives the encoding and other decoder inputs generated for the predictions (decoder outputs). We set our target output sequence during the model training as the decoder output for the model to train

against the targeted output. Figure 7-3 shows the architecture of the unidirectional LSTM.

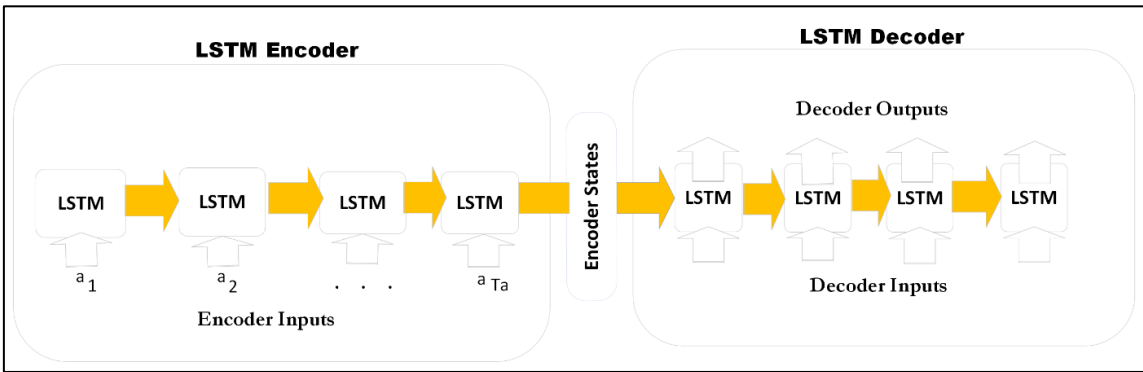


Figure 7-3 An Architecture of a Unidirectional LSTM

Algorithm 7-1 Proposed BiLSTM Approach Pseudocode	
Input:	Dataset D_{train} , D_{test} of data
Output:	Flight classification prediction results R of test data
1	Load input dataset
2	Split dataset into D_{train} and D_{test} ; 70:30 percentage ratio
3	Split D_{train} into train and validation set for training the Algorithm
4	Dataset pre-processing of the D_{train} , D_{test} Min-Max normalisation using Equation (1)
5	BiLSTM layers for enhancing the extraction of important prediction features
6	Preventing overfitting with Dropout layers
7	Batch normalisation layer using Equations (2) to (5)
8	Dense layers
9	Fully connected layers
10	Sigmoid final layer
11	<i>For each training epoch</i>
12	Train the proposed model; $Train(P)$
13	<i>End for</i>
14	Evaluate the model D_{train} , D_{test} using performance measures

7.3.3 Model Evaluation Metrics

To measure the performance of our model in the experiment, we employed some well-known benchmark metrics for evaluating models for classification and predictive tasks. These benchmark evaluation metrics are accuracy, recall, precision, F1-score and Mathew's correlation coefficient (MCC). The accuracy is the rate of the correctness of a classifier. We then use the sum of true-negative (TN), true-positive (TP), false-positive (FP) and false-negative (FN). Thus, the ratio of records correctly predicted to the total number of records is known as accuracy, as shown in Equation 9. The rate of correctly predicted values from the positive record is the recall, also known as sensitivity or true positive rate (TPR), thus calculated as Equation (7-10). The ratio of true positive records to the predicted positive records is the precision, as shown in Equation (7-11). The F1-score is the harmonic mean of the recall and precision, and Equation (7-12) shows how to calculate the F1-score. At the same time, Mathew's correlation coefficient measures the classification quality compared with each class's recall and precision relative to each other, as shown in Equation (7-13).

Accuracy can be represented by Equation (7-9):

$$\text{Accuracy} = \frac{\sum_{a=0}^Z (TP_a + TN_a)}{\sum_{a=0}^Z (TP_a + TN_a + FP_a + FN_a)} \quad (7-9)$$

Recall can be represented by Equation (7-10):

$$\text{Recall} = \frac{\sum_{a=0}^Z TP_a}{\sum_{a=0}^Z (TP_a + FN_a)} \quad (7-10)$$

Precision can be represented by Equation (7-11):

$$\text{Precision} = \frac{\sum_{a=0}^Z TP_a}{\sum_{a=0}^Z (TP_a + FP_a)} \quad (7-11)$$

F1-score can be represented by Equation (7-12):

$$\text{F1} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (7-12)$$

Mathew's correlation coefficient (MCC) can be represented by Equation (7-13):

$$\text{MCC} = \frac{\sum_{a=0}^Z (TP_a \times TN_a) - (FN_a \times FP_a)}{\sum_{a=0}^Z (TP_a + FP_a) + (TP_a + TN_a) + (TN_a + FP_a) + (TN_a + FN_a)} \quad (7-13)$$

The model results can be visualised in Confusion Matrix format, as shown in Figure 7-4.

		Predicted	
		Delayed	Not Delayd
Actual	Delayed	TP	FN
	Not Delayed	FP	TN

Figure 7-4 The Structure of a Confusion Matrix for a Binary Classification Task

Figure 7-4 is the 2X2 matrix called the Confusion Matrix, where all the correctly classified results are diagonal. The sum of the diagonals is the number of all those correctly classified.

7.4 Results and Discussion

We presented the proposed method results in this section validated on three months public dataset. The performance of our proposed method with the BiLSTM is then compared with unidirectional LSTM, and different merge mode performances was also evaluated to discover the most suitable.

We plotted a correlation matrix to show the collinearity of the multiple variables in the dataset. This helps to determine which variables have the same effect on the output variable: the flight delay. Figure 7-5 shows that Actual Elapsed Time and Distance are highly correlated, so dropping either from the dataset will not affect the model prediction. From the heap, we can see the importance of each feature in the flight delay class. The ScheduleDepTime has a high contribution

base on the correlation values on the delay class, while DepDelay and ArrDelay features have a similar level of importance on the delay classes. The month is the next most important feature of the delayed classes. Finally, distance is the next important feature, and taxiOut has the minimum importance to the delay classes.

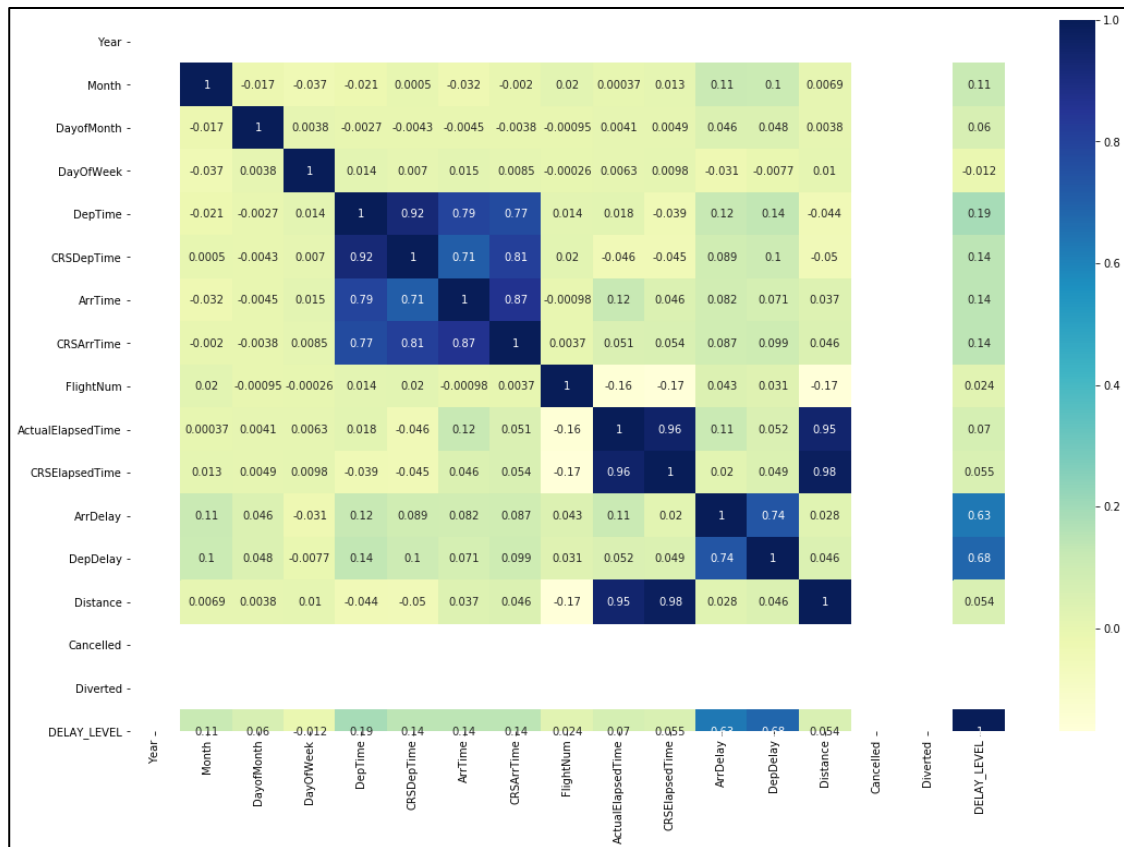


Figure 7-5: Flight Correlation Matrix

7.4.1 Experimentation Setup

We evaluated the proposed model efficacy in this section by conducting our experiment on a US BTS for 2013 with three months of records of real flights. All the computations were conducted on a Personal Computer (PC) with Intel(R) Core (TM) i7-9700 CPU with a processor speed of 3.00GHz and 32GHz RAM. We used the libraries TensorFlow Core-2.4.1, TensorFlow GPU-2.4.1, Pytorch 1.9.1, NumPy-1.19.1, pandas-0.25.3, sci-kit learn-0.23.2, Scipy-1.5.2, PySimpleGUI-4.29.0 and Matplotlib-3.3.1.

7.4.2 Results Comparative Discussions

This section presents the comparative analysis of the evaluated methods in this research. Table 7-4 contains the performance values of each of the methods.

Table 7-4 Performance comparison of LSTM and BiLSTM models

S/No	Methods	Classes	Precision	Recall	F1-Score	Support	MCC
1	LSTM	Class 0	0.8384	0.9443	0.8756	107892	0.4644
		Class 1	0.4469	0.0743	0.0896	101823	
			-	-	0.7645	209715	
		Accuracy					
		Macro Average	0.4532	0.4532	0.4356	209715	
		Weighted Average	0.7362	0.7453	0.7453	209715	
2	BiLSTM	Class 0	0.8324	0.9898	0.8945	107892	0.9944
		Class 1	0.5643	0.0989	0.4988	101823	
			-	-	0.9756	209715	
		Accuracy					
		Macro Average	0.4202	0.4332	0.4122	209715	
		Weighted Average	0.70023	0.7234	0.7213	209715	

The LSTM model had an accuracy of 76.45%, which is comparatively lower than the BiLSTM model. The LSTM model has a precision, recall and F1-Score of 83.84, 94.43% and 87.56%, respectively, in predicting class 0, which is better than that of class 1, having a precision of 44.69%, recall of 7.43%, F1-Score of 8.96%, respectively. It can be seen in the confusion matrix classification report presented in Figure 7-6, where the model classified all the delay classes without misclassification. In contrast, the not delayed class has some misclassification. When predicting classes of the delay, the LSTM model results show a lower performance when compared with the BiLSTM; however, a better performance LSTM model shows on the precision of class 1, which indicates the correctly retrieved instances of the model on the class. The mutual correlation of the LSTM predicted classes, an MCC score of 0.4644, is significantly lower than the BiLSTM because of their overall differences in the accuracy of predictions, which means forward and backward networks of BiLSTM help in retrieving the sequences of the previous layers during the training.

The results show that the BiLSTM model performs better than the LSTM in prediction instances with just a few misclassifications, as can be seen in the Confusion matrix in Figure 7-7. The results of the BiLSTM model show an accuracy of 97.56%, an increase of 21.11% compared to the LSTM accuracy model. The model performance in class 0 is solid, with precision, recall and F1-Score of 83.24%, 98.98% and 89.45%, respectively. Although the performance has a small margin compared to LSTM on class 0, the model correctly identifies a higher proportion of the not delay class. The precision of class 1 is 56.43%, but it is better than the baseline LSTM model in class 1. The models recall and F1-Score in class 1 are 9.89% and 49.88%, respectively. The MCC score for the BiLSTM model is 0.9944, which shows a high correlation between the two predicted classes of the model and, thus, means a good model.

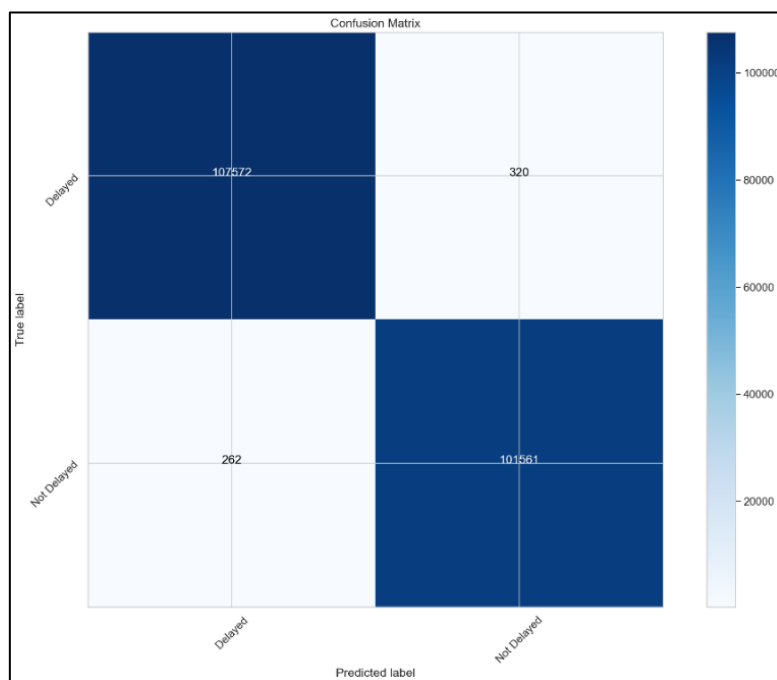


Figure 7-6: Confusion Matrix for LSTM approach

The subsequent data points are more important than the previous ones, which is considered missing information by the LSTM model. The BiLSTM model solves this LSTM model problem by doing two forward and backward LSTM model training for the training dataset sequences. The Confusion matrix in Figure 7-6 shows that the unidirectional LSTM model correctly predicts 107,572 and 101,561 of the delayed and not-delayed flight classes, respectively, while Figure

7-7 shows that BiLSTM correctly predicted 107, 892 and 101, 580 of the delayed and not-delayed flights classes. It clearly shows from the Confusion matrix results that the BiLSTM model's performance in predicting the flight is better than that of the LSTM model. The overall result shows the advantage of using deep BiLSTM over an LSTM because the prediction error of an LSTM tends to increase significantly as the number of prediction steps.

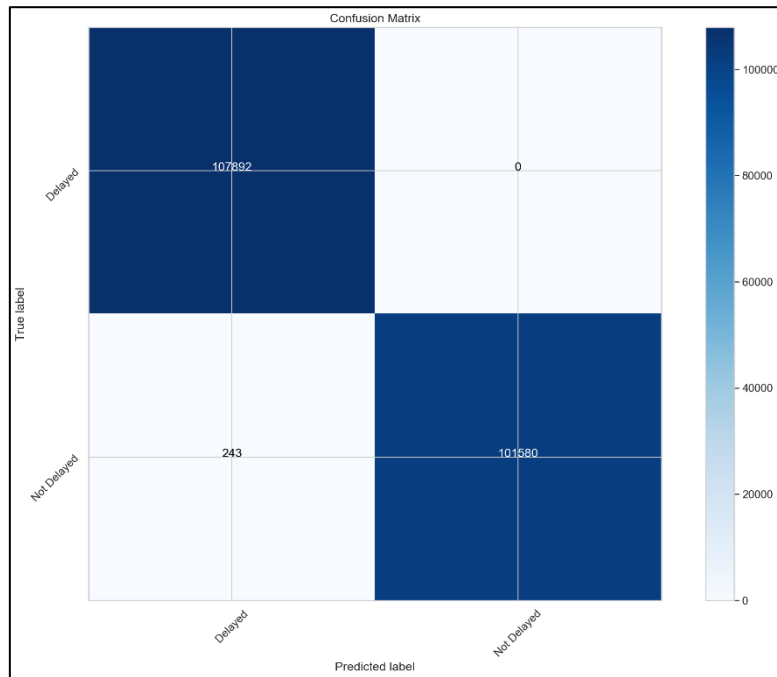


Figure 7-7: Confusion Matrix for BiLSTM approach

7.5 Conclusion and Future Direction

The delay propagation in the air traffic network propagates through the entire network with speed due to the complex nature of the system. In this research, we compared two classification and prediction methods known as LSTM and BiLSTM by building a classification model to analyse flight delay on-time datasets to assess the efficacy and viability of BiLSTM to LSTM. The BiLSTM model has at highest accuracy of 97.56% than the unidirectional LSTM with an accuracy of 76.45%; the change in the accuracy of the BiLSTM model shows its potential in the prediction of flight delays. BiLSTM plausibly outperforms the LSTM model due to its bi-directional (forward and backwards), leveraging any feature selection capacity provided by the bi-directional ability in the architecture to handle missing sequences during the model training. Another reason could be that BiLSTM takes

additional time to fetch batches of data to have an equilibrium, though it is slow because of this process. It indicates that the additional features associated with the data are captured by BiLSTM but cannot be exposed by the unidirectional LSTM models because the training is only one way. Thus, we recommend BiLSTM over LSTM for binary time series analysis, classification, or prediction.

This paper has contributed to the vast area of discourse in air transportation and stakeholders in decision-making, thereby improving air passengers' or customers' experience and increasing income from aviation and non-aviation services. In the future, we recommend more cross-validation methods and larger sample sizes across different regions to develop the models and further evaluate the regional performance of the model towards a universal central prediction of flights across nations. Other factors that can be improved regarding the neural network aspect should be architectural design to achieve better tuning and higher accuracy of the models by training the model with the weather, aircraft age, aircraft model, and factors limiting airport infrastructure (i.e., available runways to flights). It may help solve the low performance related to the unidirectional LSTM model because, from our dataset, the weather-related delay is highly imbalanced.

Acknowledgement and funding

The Petroleum Trust Development Fund (PTDF) Nigeria partially funded this work through grant number PTDF/ED/OSS/PHD/DBB/1558/19 for one of the first author's PhD studies. We also thank the UKRI for the Covid-19 recovery grant under the budget code SA077N.

Data availability

All data used in this research is available at <https://www.bts.gov/>.

7.6 References

1. Cheevachaipimol W., Teinwan B., Chutima P. Flight delay prediction using a hybrid Deep Learning method. *Engineering Journal*. 2021; 25(8): 99–112. Available at: DOI:10.4186/ej.2021.25.8.99
2. Wu Y., Mei G., Shao K. Revealing Influence of Meteorological Conditions and Flight Factors on Delays Using XGBoost. *Journal of Computational Mathematics and Data Science*. Elsevier B.V.; 2022; : 100030. Available at: DOI:10.1016/j.jcmds.2022.100030
3. Bisandu DB., Moulitsas I., Filippone S. Social ski driver conditional autoregressive-based Deep Learning classifier for flight delay prediction. *Neural Computing and Applications*. Springer London; 2022; 34(11): 8777–8802. Available at: DOI:10.1007/s00521-022-06898-y
4. Civil Aviation Administration of China (CAAC). Statistics of Key Performance Indicators for China ' s Civil Aviation Industry in December 2022. 2022.
5. Fu X., Lei Z., Liu S., Wang K., Yan J. On-time performance policy in the Chinese aviation market - An innovation or disruption? *Transport Policy*. Elsevier Ltd; 2020; 95(June): A14–A23. Available at: DOI:10.1016/j.tranpol.2020.06.008
6. Liu YJ., Cao WD., Ma S. Estimation of arrival flight delay and delay propagation in a busy hub-airport. *Proceedings - 4th International Conference on Natural Computation, ICNC 2008*. 2008; 4: 500–505. Available at: DOI:10.1109/ICNC.2008.597
7. Maxson RW. Prediction of Airport Arrival Rates Using Data Mining Methods. 2018;
8. Carvalho L., Sternberg A., Maia Gonçalves L., Beatriz Cruz A., Soares JA., Brandão D., et al. On the relevance of data science for flight delay research: a systematic review. *Transport Reviews*. Taylor & Francis; 2021; 41(4): 499–528. Available at: DOI:10.1080/01441647.2020.1861123
9. Ball M., Barnhart C., Dresner M., Hansen M., Neels K., Odoni A., et al. Total delay impact study: a comprehensive assessment of the costs and impacts of flight delay in the United States. *Nextor*. 2010; : 5–14, 25–30. Available at: http://www.isr.umd.edu/NEXTOR/pubs/TDI_Report_Final_10_18_10_V3.pdf
10. Chen Y., Yu J., Tsai SB., Zhu J. An Empirical Study on the indirect impact of flight delay on China's economy. *Sustainability (Switzerland)*. 2018; 10(2). Available at: DOI:10.3390/su10020357
11. Efthymiou M., Njoya ET., Lo PL., Papatheodorou A., Randall D. The impact of delays on customers' satisfaction: An empirical analysis of the british airways on-time performance at heathrow airport. *Journal of Aerospace Technology and Management*. 2019; 11: 1–13. Available at: DOI:10.5028/jatm.v11.977
12. Gu Y., Yang J., Wang C., Xie G. Early warning model for passenger disturbance due to flight delays. *PLoS ONE*. 2020; 15(9 September): 1–13. Available at: DOI:10.1371/journal.pone.0239141
13. Wu CL. Monitoring aircraft turnaround operations - Framework development, application and implications for airline operations.

- Transportation Planning and Technology. 2008; 31(2): 215–228. Available at: DOI:10.1080/03081060801948233
14. Cai KQ., Zhang J., Xiao MM., Tang K., Du WB. Simultaneous optimisation of airspace congestion and flight delay in air traffic network flow management. *IEEE Transactions on Intelligent Transportation Systems*. 2017; 18(11): 3072–3082. Available at: DOI:10.1109/TITS.2017.2673247
 15. Etani N. Development of a predictive model for on-time arrival flight of airliner by discovering correlation between flight and weather data. *Journal of Data*. 2019; 6(1): 1–7. Available at: DOI:10.1186/s40537-019-0251-y
 16. Gui G., Liu F., Sun J., Yang J., Zhou Z., Zhao D. Flight delay prediction based on aviation data and Machine Learning. *IEEE Transactions on Vehicular Technology*. 2020; 69(1): 140–150. Available at: DOI:10.1109/TVT.2019.2954094
 17. Qu J., Zhao T., Ye M., Li J., Liu C. Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data. *Neural Processing Letters*. Springer US; 2020; 52(2): 1461–1484. Available at: DOI:10.1007/s11063-020-10318-4
 18. Ding Y. Predicting flight delay based on multiple linear regression. *IOP Conference Series: Earth and Environmental Science*. 2017; 81(1): 0–7. Available at: DOI:10.1088/1755-1315/81/1/012198
 19. Dou X. Flight Arrival Delay Prediction And Analysis Using Ensemble Learning. 2020; (It nec): 836–840. Available at: DOI:10.1109/itnec48623.2020.9084929
 20. Baumgarten P., Malina R., Lange A. The impact of hubbing concentration on flight delays within airline networks: An empirical analysis of the US domestic market. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier Ltd; 2014; 66: 103–114. Available at: DOI:10.1016/j.tre.2014.03.007
 21. Kim YJ., Choi S., Briceno S., Mavris D. A Deep Learning approach to flight delay prediction. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. IEEE; 2016; 2016-Decem: 1–6. Available at: DOI:10.1109/DASC.2016.7778092
 22. Lin Y., Zhang J wei., Liu H. Deep Learning based short-term air traffic flow prediction considering temporal–spatial correlation. *Aerospace Science and Technology*. Elsevier Masson SAS; 2019; 93: 105113. Available at: DOI:10.1016/j.ast.2019.04.021
 23. Mueller ER., Chatterji GB. Analysis of aircraft arrival and departure delay characteristics. *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*. 2002. pp. 1–14. Available at: DOI:10.2514/6.2002-5866
 24. Vandehezad M., Holmgren J. Efficient flight schedules with utilising Machine Learning prediction algorithms. 2020;
 25. Belcastro L., Marozzo F., Talia D., Trunfio P. Using scalable data mining for predicting flight delays. *ACM Transactions on Intelligent Systems and Technology*. 2016; 8(1). Available at: DOI:10.1145/2888402
 26. Chen J., Li M. Chained predictions of flight delay using Machine Learning. *AIAA Scitech 2019 Forum*. 2019. pp. 1–25. Available at: DOI:10.2514/6.2019-1661

27. Ye B., Liu B., Tian Y., Wan L. A methodology for predicting aggregate flight departure delays in airports based on supervised learning. *Sustainability*. 2020; 12(7): 2749. Available at: DOI:10.3390/su12072749
28. Bisandu DB., Homaid MS., Moulitsas I., Filippone S. A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective. *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK*. ACM; 2021. pp. 1–10. Available at: DOI:https://doi.org/10.1145/3505711.3505712
29. Manna S., Biswas S., Kundu R., Rakshit S., Gupta P., Barman S. A statistical approach to predict flight delay using gradient boosted decision tree. *ICCIDS 2017 - International Conference on Computational Intelligence in Data Science, Proceedings*. 2018; 2018-Janua: 1–5. Available at: DOI:10.1109/ICCIDS.2017.8272656
30. Chakrabarty N. A data mining approach to flight arrival delay prediction for American airlines. *IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference*. 2019. pp. 102–107. Available at: DOI:10.1109/IEMECONX.2019.8876970
31. Kuhn N., Jamadagni N. Application of Machine Learning Algorithms to Predict Flight Arrival Delays. *Cs229*. 2017; : 1–6.
32. Takeichi N., Kaida R., Shimomura A., Yamauchi T. Prediction of delay due to air traffic control by Machine Learning. *AIAA Modelling and Simulation Technologies Conference*, 2017. 2017; (January): 1–8. Available at: DOI:10.2514/6.2017-1323
33. Gopalakrishnan K., Balakrishnan H. A comparative analysis of models for predicting delays in air traffic networks. *12th USA/Europe Air Traffic Management R and D Seminar*. 2017;
34. Yu B., Guo Z., Asian S., Wang H., Chen G. Flight delay prediction for commercial air transport: A Deep Learning approach. *Transportation Research Part E: Logistics and Transportation Review*. Elsevier; 2019; 125(March): 203–221. Available at: DOI:10.1016/j.tre.2019.03.013
35. Khanmohammadi S., Tutun S., Kucuk Y. A New Multilevel Input Layer Artificial Neural Network for Predicting Flight Delays at JFK Airport. *Procedia Computer Science*. The Author(s); 2016; 95(November): 237–244. Available at: DOI:10.1016/j.procs.2016.09.321
36. Lv Y., Duan Y., Kang W., Li Z., Wang FY. Traffic Flow Prediction with Data: A Deep Learning Approach. *IEEE Transactions on Intelligent Transportation Systems*. IEEE; 2015; 16(2): 865–873. Available at: DOI:10.1109/TITS.2014.2345663
37. Zhang H., Song C., Wang H., Xu C., Guo J. Airport Delay Prediction Based on Spatiotemporal Analysis and Bi-LSTM Sequence Learning. *Proceedings - 2019 Chinese Automation Congress, CAC 2019*. 2019; (2017): 5080–5085. Available at: DOI:10.1109/CAC48633.2019.8996754
38. Karim F., Majumdar S., Darabi H., Chen S. LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access*. 2017; 6: 1662–1669. Available at: DOI:10.1109/ACCESS.2017.2779939
39. United States Department of Transportation. Bureau of Transportation Statistics, Bureau of Transportation Statistics. United States Department of

- Transportation. 2020. Available at: <https://www.bts.gov/> (Accessed: 3 March 2020)
- 40 Yazdi MF., Kamel SR., Chabok SJM., Kheirabadi M. Flight delay prediction based on Deep Learning and Levenberg-Marquart algorithm. *Journal of Data*. Springer International Publishing; 2020. Available at: DOI:10.1186/s40537-020-00380-z
 - 41 Cios KJ., Pedrycz W., Swiniarski RW., Kurgan LA. Data mining: A knowledge discovery approach. *Data Mining: A Knowledge Discovery Approach*. 2007. pp. 1–606. Available at: DOI:10.1007/978-0-387-36795-8
 42. Felix A. Gers; Jürgen Schmidhuber; Fred Cummins. Learning to forget: Continual prediction with LSTM. *Neural computation*. 2000; 12(10): 2451–2471. Available at: DOI:10.1162/089976600300015015
 43. Hochreiter S., Jürgen Schmidhuber J. Long Shortterm Memory. *Neural Computation*. 1997; 9(8): 17351780.
 44. Graves A., Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*. 2005; 18(5–6): 602–610. Available at: DOI:10.1016/j.neunet.2005.06.042

Chapter 8: Social Ski Driver Conditional Autoregressive-Based Deep Learning Classifier for Flight Delay Prediction

The air transportation industry's importance of robust flight delay prediction has recently increased. This industry seeks alternative methods and technologies for more robust flight delay prediction because of its significance for all stakeholders. The most affected are airlines that suffer from monetary and passenger loyalty losses. Several studies have attempted to analyse and solve flight delay prediction problems using Machine Learning methods. This research proposes a novel alternative method: Social Ski Driver Conditional Autoregressive-based (SSDCA-based) Deep Learning. Our proposed method combines the Social Ski Driver algorithm with Conditional Autoregressive Value at Risk by Regression Quantiles. We consider the most relevant instances from the training dataset, which are the delayed flights. We applied data transformation to stabilise the data variance using Yeo-Johnson. We then train and test our data using Deep Recurrent Neural Network (DRNN) and SSDCA-based algorithms. The SSDCA-based optimisation algorithm helped us choose the right network architecture with better accuracy and less error than the existing literature. The results of our proposed SSDCA-based method and existing benchmark methods were compared. Our proposed method's efficiency and computational time are compared against the existing benchmark methods. The SSDCA-based DRNN provides a more accurate flight delay prediction with 0.9361 and 0.9252 accuracy rates on both dataset-1 and dataset-2, respectively. To show the reliability of our method, we compared it with other meta-heuristic approaches. The result is that the SSDCA-based DRNN outperformed all existing benchmark methods tested in our experiment.

8.1 Introduction

The civil aviation sector is a distributed network of large interconnected elements designed to meet the common aim of on-time air transportation and passengers' expectations [1,2]. Because of flight connectivity, flight delays at airports, especially for commercial hub airports, usually propagate to other individual airports or even to the entire air transportation network.

Without proper monitoring and control, such delays can expand over time, resulting in poor airport performance and causing unnecessary dissatisfaction for passengers [1, 14]. Over the last few years, this sector has rapidly grown in areas such as customers, infrastructure and territorial coverage [3, 4]. A statistical report from the International Air Transport Association (IATA) reveals that in 2012, airlines' worldwide base incurred over \$160 billion in total energy costs [5, 6, 7]. With the rapid growth of Communication, Navigation and Surveillance (CNS), increased air traffic and airspace capacities, flight schedule planners try to reduce buffer time among flight departures and arrivals to maximise aircraft utilisation.

The air transportation network architecture is tight and complex, making it vital to develop accurate prediction models critical for intelligent aviation systems [68]. It is necessary to predict flight departure or arrival delays with high accuracy. Passenger anxiety can be avoided by efficiently arranging their schedules to access a prediction model for a specific airport taxi time, with an explanatory variable being computed before prediction [68, 70, 71]. Additionally, airport management aims to provide better service with improved airport gate availability [3, 8, 9]. Increased air traffic and flight delays have become a severe and prominent global issue [1]. Based on the United States (US) Bureau of Transportation Statistics (BTS) report, most airline flights arrived 15 minutes late at their destination [10, 11, 12, 66, 76, 77, 78, 79, 80], thus incurring a loss of \$30 billion, which is a challenge to the air transportation system [10, 13]. Delay analyses have become an important research topic, and delay prediction has been the subject of earlier studies [10, 19, 20]. As stated earlier, delays have a significant financial impact. Therefore, it is important to introduce intelligent systems to automate airports, passengers, and commercial airlines' monitoring and decision processes [23]. Highly accurate predictions and real-time monitoring systems are indispensable tools to that effect [1, 18, 21, 22, 23, 24, 68, 69, 70, 71, 72]. Proposed strategies include Collaborative Decision Making (CDM) [3, 26], Ground Delay Programs (GDP) [3, 27], and Air Traffic Flow Management (ATFM) [3, 28] to improve the information flow among participating airports. A few years ago, the research methodologies utilised for predicting delay propagation

were from statistical, network theory, Machine Learning and agent-enabled methods [3, 29].

Comparing statistical approaches with Machine Learning methods has become popular in recent years. The field of transportation systems and aerospace research has experienced a significant number of Machine Learning models, such as K-nearest neighbour (k-NN), support vector machine (SVM), and artificial neural network (ANN) models [68]. Some studies seek to improve the model prediction performance by introducing variant neural network models [73, 74, 75]. Using only Machine Learning methods on historical data without an optimisation algorithm has proven ineffective. In contrast, our proposed method uses historical data from two datasets with a novel optimisation algorithm. The authors [10, 30] employed the traditional statistical approach to characterise and distribute flight delays. In [10, 31], the authors combine terminal airports' weather forecasts, convective weather forecasts, and the scheduled flights for predicting daily airport delay time in terms of the Weather Impacted Traffic Index (WITI) metric. As reported in the 2017 BTS report, only 0.72% of flight delays were attributed to extreme weather [10]. Deep Learning classifiers are the most broadly used traffic prediction techniques [15, 33, 34, 35]. The Deep Learning techniques come under supervised and unsupervised Machine Learning algorithms [15, 32, 36, 37, 38, 39, 60, 73, 74, 75]. None of the previous studies has taken multiple routes into full account in the prediction. In contrast, we introduce a feature fusion method that utilises the complete flight information on different routes and combines them to improve performance. In our studies, we will focus on non-weather-induced delays.

Our paper aims to propose a novel flight delay prediction strategy that utilises Social Ski Driver Conditional Autoregressive-based (SSDCA-based) Deep Long Short-Term Memory (LSTM). We conduct data pre-processing initially to improve the data quality; then, we perform the data transformation based on the Yeo-Johnson transformation for further data processing. Yeo-Johnson transformation transfers the data with no loss of its original quality. It works like Box-cox transformation, but data values must not be strictly positive, and it is advantageous over other transformation techniques. Also, it makes the data

distribution more symmetric, thereby handling any skewness from the datasets. We then perform feature fusion using the Deep Recurrent Neural Network (Deep RNN) for fusing the imperative features. Here, we train the Deep RNN using the developed SSDCA, improving the model learning process. Finally, we perform flight delay prediction using the Deep LSTM. Furthermore, we compared the newly developed SSDCA with other optimisation algorithms such as Social Ski Driver (SSD), Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO), Honey-Bee Optimisation (HBO) and Earthworm Optimisation Algorithm (EWA) [40, 41]. The accuracy (AC) of the SSDCA outperforms the other existing methods. It is also worth mentioning that the proposed method's computational time is less than the other methods. In terms of error rate, such as Root Mean Square Error (RMSE), Mean Square Error (MSE) and Mean Absolute Error (MAE), the proposed SSDCA outperformed other methods.

The contribution of the paper is:

- i. **Proposed SSDCA enabled Deep LSTM for flight delay prediction:** We introduced the classifier; SSDCA algorithm drove Deep LSTM by modifying the training process of the Deep LSTM with SSDCA algorithm newly proposed by incorporating SSD with CAViaR for biases and weights optimal tuning. We utilise the Deep RNN for feature fusion, which is trained by the proposed SSDCA. Also, we adapted the SSDCA-enabled Deep LSTM for predicting flight delays.

The rest of this paper is organised as follows: Section 8.2 describes the conventional flight delay prediction strategies employed in the literature and the challenges that inspire the development of the novel technique. Section 8.3 describes the proposed model for flight delay prediction based on the SSDCA-based Deep Learning classifier. Section 8.4 presents our model results and compares them with results from existing methods. Finally, Section 8.5 contains the results and a discussion of the findings to conclude with possible future directions.

8.2 Related Literature

The vast volume of collected data from the commercial aviation system makes developing Machine Learning and Artificial Intelligence algorithms a popular

candidate approach in predicting flight delays. Traditional methods such as support vector machine, neural network, fuzzy logic, tree-based methods and K-nearest neighbour are the most common data-driven methods [2]. Güvercin et al. [37] proposed a clustered airport modelling approach for forecasting flight delays using airport networks. The method provided accurate forecasts for flight delays. However, during the training, the method uses only a few samples, which harms the model's prediction performance. Lambelho et al. [38] assessed airports' generic strategic schedules using flight cancellation and delay predictions. The method's performance was good in cancellations and delayed flight departures but did not consider other features such as origin and destination to improve the predictions. Tu et al. [23] studied the factors causing major departure delays at Denver International Airport and the departure delay distribution using a probabilistic approach for United Airlines. The study attempts to separate contributing factors but focuses mainly on a single airport and does not consider the network effect. Pathomsiri et al. [39] assessed US joint production on-time and delay performance using a non-parametric function approach.

Some researchers apply operational research, simulation, queueing theory and optimisation to simulate flight delays for an optimised policymakers' system. Pyrgiotis et al. [3] studied an extensive network of delay propagation in major airports in the USA through network decomposition and an analytical queueing model. Ankan et al. [40] analyse delay propagation through air traffic networks with empirical data by developing a stochastic model. A Bayesian network method for estimating delay propagation considers the element-oriented and complex network distribution properties in three commercial aviation systems in the USA. These methods are valuable in understanding interactions and root causes amongst delay occurrence elements. However, for the individual flights, these models did not yield sufficiently accurate predictions [41]. Rebollo et al. [42] predict departure delays by adopting random forest algorithms using air traffic characteristics as input features. When predicting departure delay for a two-hour forecast window, the model had an error of 21 minutes. Choi et al. [43] employ several machine-learning algorithms and combine weather forecasts with flight schedules to predict scheduled times. Perez-Rodriguez et al. [44] proposed a

model to predict daily aircraft delay probabilities in arrivals using asymmetric logic probability.

Recently, Deep Learning algorithms have been employed to improve the accuracy of flight delay prediction. Yu et al. [1] study flights at Beijing International Airport using a novel Deep Believe Network with a support vector regression method (SVR) to analyse high-dimensional data. The model achieves a Mean Absolute Error (MAE) of 8.41 minutes with high accuracy, but this study was limited to a single airport, and the propagation effects were not evaluated. Kim et al. [45] predict departure and arrival flight delays using a recurrent neural network (RNN) of an individual airport with a day-to-day sequence. Their study shows that a more in-depth architecture improved the accuracy of the RNN. However, the model can only perform a binary delay prediction and does not quantify its magnitude. Chen and Li [11] developed a Machine Learning method for chained predictions of flight delays. The method provides an averagely reasonable, accurate and practical result for delay prediction but did not include an adequately dataset which could have improved the accuracy. Ai et al. [26] developed Convolutional LSTM for temporal and spatial distribution flight delay prediction in the network. The method achieved good classification accuracy but did not include human factors.

Guleria et al. [4] presented a Multi-Agent Approach for reactionary flight delay prediction. The method helps in the flight scheduling system by identifying itineraries. However, the technique did not test delay propagation trees for better performance. Chen et al. [6] presented the Information Gain-Support Vector Machine method for determining how improvements in flight delays of the studied airlines based in China can reduce CO₂ emissions. The proposed approach reduces the limitations of the traditional data envelopment analysis (DEA) model. However, the authors focus on a limited number of airlines and recommend further experiments to confirm the method's validity. Maryam et al. [46] proposed a model for predicting flight delays based on the Levenberg-Marquart algorithm and Deep Learning. The accuracy of the proposed model in forecasting flight delays was good. However, the results show that the imbalanced form's standard deviation is higher than all balanced evaluation parameters. Balanced data tends

to lead to a lower standard deviation (SD). Ehsan and Seyedmirsajad [47] proposed an approach to predict and analyse departure flight delays using MATLAB R2018b [67] SVM implementations. The author recommended expanding the research to include National Aviation System (NAS)-wide airports in the analysis for more complete results. Daniel et al. [48] proposed using a supervised neural network to predict air traffic delays with multilevel input layers. The primary aim is to present a prediction model for delays in the air route by applying artificial neural networks (ANN) during the model training; authors concluded that parameters such as the day of the week, the block hour, or the airline had a stronger influence than meteorology on the delay.

Regardless of the accuracy of the models, our observation of the methods used in the above works is that their training phase is slow, a characteristic that can become a limiting factor when the size of the data set grows. Another aspect of the training phase is the existence of outliers in the data. Such outliers could exist, for example, because of extreme delays that are not often encountered. Yu et al. [1] eliminate extreme value delays from the bottom and the top 1% as outliers. Tu et al. [23] reduce the smoothing spline approach's influence by excluding extreme data preparation observations. The authors in [11] and [49] adopted a random forest method because of its low sensitivity to outliers. This average optimisation technique was unsuitable for our study because it may eliminate essential features through manual selection.

Accurate prediction of flight delay propagation at the national network level is essential, especially for flights with reoccurring delays, to reduce unwanted expenses. Based on the recent state-of-the-art architectural model of RNN units, LSTM has been promising in addressing this limitation. Researchers have applied it in several fields, including predicting traffic, because of the ability to learn time series features and temporal correlation. Ma et al. [50] performed traffic speed prediction using LSTM on historical microwave detectors' speed data and compared their approach with other approaches regarding stability and accuracy. Liu et al. [51] proposed a generative Deep Learning model comprising LSTM encoder and decoder layers to predict aircraft trajectories. The LSTM shows it can perform good feature extraction and learn useful temporal correlation

features [52]. However, despite all these advancements, there is still a need to improve prediction performance and reliability, potentially using an optimisation algorithm to generate more efficient and accurate models.

Challenges

We elaborated on some issues faced by the existing predictive approaches for flight delay as follows:

- i. The Deep Learning model and convolutional LSTM [26] were developed for predicting flight delays. However, the method failed to consider the airport's delays to optimise take-off and landing intervals.
- ii. In [38], the authors developed a generic strategic schedule assessment for predicting flight delays. However, the method does not utilise features such as unique air carriers, tail number of aircraft, and origin/destination airports with trigonometric transform function in the model training to enhance predictions' accuracy.
- iii. In [1], the authors introduced a deep belief network method to determine flight delays; the method did not use an optimisation algorithm without access to the required data on air traffic control because of confidentiality considerations for improving performance.

Our proposed method includes the departure and arrival details in the selected features. Thus, our method overcomes the challenge of take-off and landing by considering the dataset's actual and scheduled time for arrival and departure features. Our proposed method also uses categorical and numerical data features to enhance prediction performance.

8.3 Problem Statement

Flight delays may occur because of several unforeseen events, affecting airlines, airports and passengers. Developing more accurate models for predicting flight delays has become essential because of the rapid increase in flight complex data overflow, the limited number of prediction methods, and the air transportation system network's complexity. The proposed method builds an accurate flight delay prediction model in this context. We assume the input data, represented by D , to be a collection of $D_{i,j}$'s, as defined in Equation (8-1).

$$K = \{D_{i,j}\}; (1 \leq i \leq N'); (1 \leq j \leq T') \quad (8-1)$$

where N' is the total number of data points and T' is the total number of attributes.

Hence, $D_{i,j}$ represents data in a database K depicting j^{th} mixed attributes of i^{th} data. Thus, the expression $[N' \times T']$ denotes the size of the input data, indicated by $(D)_{N' \times T'}$. We will be extracting 6 features f_i from the input data D and fuse them into f as shown in Equation (8-2) before feeding them as input to the Deep Learning classifier for predicting the flight delay as a final output O_f .

$$f = f_i; \quad i = 1 \text{ to } 6 \quad (8-2)$$

8.4 Flight Delay Prediction using the Proposed Social Ski Driver Conditional Autoregressive-based (SSDCA) Deep Learning classifier

This section explains the proposed SSDCA-based Deep Learning classifier for predicting flight delay. The steps followed in the developed model are pre-processing, data transformation, feature fusion, and prediction. We feed the pre-processing module with the input data and then transmit data using the Yeo-Johnson transformation [53]. Afterwards, we forward the transformed data to the feature fusion module, where Deep RNN performs the feature fusion [54]. We select an optimal number of layers in the Deep RNN method using the proposed SSDCA-based algorithm. The newly developed SSDCA algorithm combines the SSD [58] and CAViaR [56]. Finally, we perform the prediction using the Deep LSTM method [57] that we trained using the developed SSDCA method. Figure 8-1 shows the flight delay detection block diagram using the newly proposed SSDCA-based Deep Learning.

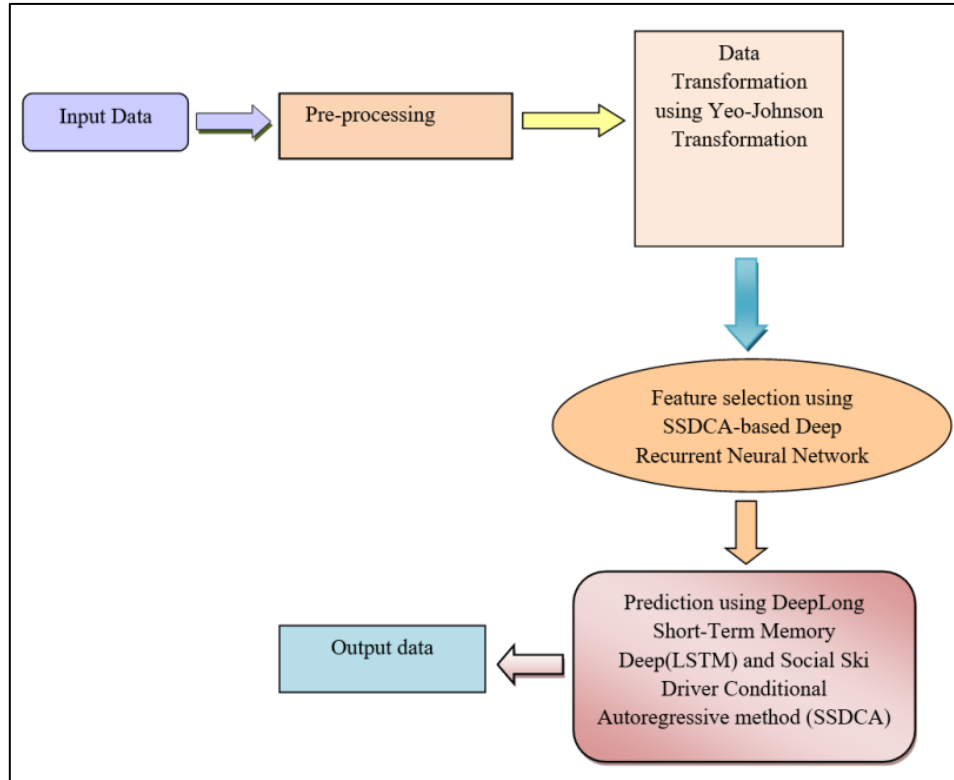


Figure 8-1 Block diagram of the Flight Delay Prediction using the Proposed SSDCA-based Deep Learning Classifier

8.4.1 Dataset Description

The dataset used in this paper is from the US flight data downloaded from the US Government BTS for January and February 2019 and 2020, respectively [61, 62]. The extracted data feature contains flight information with 21 features for time series analysis and flight delay predictions, as shown in Table 8-1 [66]. There are over 1,000,000 row instances of commercial flights. The dataset records are inconsistent and incomplete, with many missing duplicates and null values. We initially need to clean to make the data complete and suitable for further pre-processing by converting the attributes to the most appropriate forms for applying Deep Learning and Machine Learning methods.

Table 8-1 Datasets information

S/No	Feature Name	Data Type	Attribute Type	Brief format explanation
1	Year	int64	Numerical (Discrete)	In four digits format, e.g. 2000
2	Month	int64	Numerical (Discrete)	In two digits format, e.g. 12
3	Day of Month	int64	Numerical (Discrete)	In two digits format, e.g. 01-31
4	Day of Week	int64	Numerical (Discrete)	In one-digit format, e.g. 1 (Monday) - 7 (Sunday)
5	Departure Time	float64	Numerical (Continuous)	In four digits format, e.g. 1456
6	Scheduled Departure Time	float64	Numerical (Continuous)	In four digits format, e.g. 1456
7	Arrival Time	float64	Numerical (Continuous)	In four digits format, e.g. 1456
8	Scheduled Arrival Time	float64	Numerical (Continuous)	In four digits format, e.g. 1456
9	Unique Carrier Code	object	Categorical (Nominal)	2 letters code, e.g. PS
10	Flight Number	object	Numerical (Discrete)	Number of flights, e.g. 1454
11	Actual Elapsed Time	float64	Numerical (Continuous)	In two digits format, e.g. 93
12	Scheduled Elapsed Time	float64	Numerical (Continuous)	In two digits format, e.g. 94
13	Flight Time	float64	Numerical (Continuous)	In two digits format, e.g. 10
14	Arrival Time	float64	Numerical (Continuous)	In two digits format, e.g. 12
15	Departure Time	float64	Numerical (Continuous)	In two digits format, e.g. 11
16	Origin	object	Categorical (Nominal)	3 letters code eg. LHR
17	Destination	object	Categorical (Nominal)	3 letters code, e.g. MAN
18	Distance	int64	Numerical (Continuous)	The miles covered, e.g. 1448
19	Diverted	int64	Binary (Categorical)	In two binary format, e.g. 0 or 1
20	Carrier Delay	int64	Binary (Categorical)	In two binary format, e.g. 0 or 1
21	Weather Delay	int64	Binary (Categorical)	In two binary format, e.g. 0 or 1

Features used for the model training and testing

Several dataset features are irrelevant to our experiment, and we only kept the relevant features that have a high contribution to flight delay. We use the following features in training and testing our model:

- (i) Flight Date: the date on which the flight was performed.
- (ii) Origin: departure airport.
- (iii) Destination: arrival airport.
- (iv) Departure delay: The difference between actual and scheduled departure times.
- (v) Arrival delay: The difference between actual and scheduled arrival times.
- (vi) Distance: the miles covered by the flight.

The departure and arrival delays are logically highly correlated, and any experience of delay on the departure flight will certainly affect the flight arrival. The authors in [19] have proved that the departure airport largely causes the origin of congestion at the destination airport. For this reason, we have the selected features.

8.4.2 Pre-processing

The input data is a collection of categorical, numerical and time attributes. Categorical attributes indicate the airport station, for example, Heathrow, Delhi, Chennai etc. Numerical attributes represent the path values, humidity, etc. Finally, time attributes signify the date and time stamp for departure and arrival. Initially, we convert the categorical attributes to categorical numbers, assigning the airport stations with a unique identifier number 1, 2, 3, etc. After that, we change both the numerical and time attributes from text to number data. Once we get the pre-processed data, we compute the missing value using the average and frequency methods for numerical and categorical attributes. Thus, the pre-processed data output is indicated $(P)_{N \times T}$ where the dimension of pre-processed data is $[N \times T]$. Figure 8-2 shows the determination of pre-processed data output.

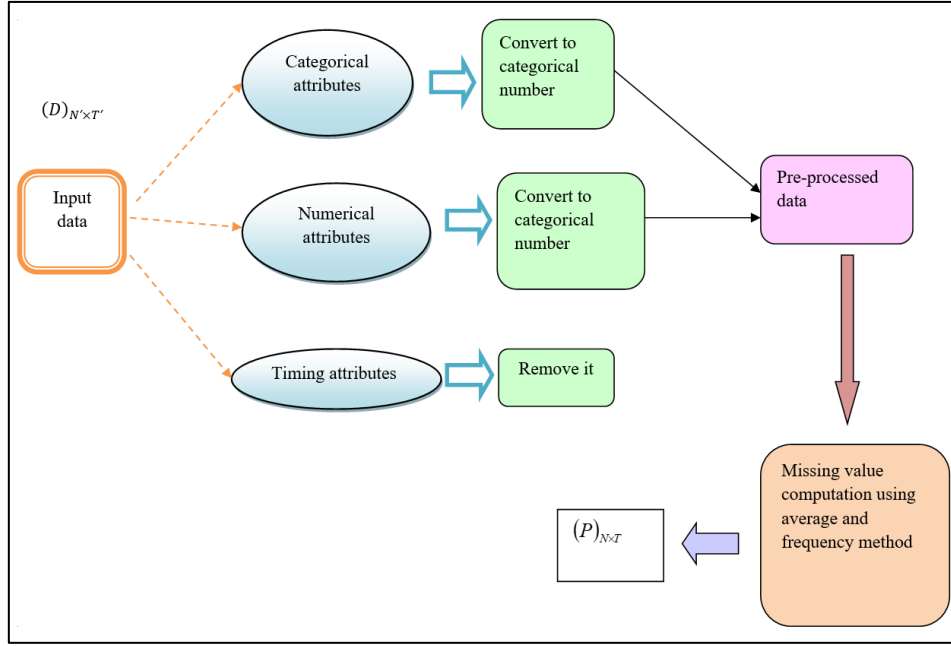


Figure 8-2 Schematic Diagram for the Determining-processing Output

8.4.3 Data Transformation using Yeo-Johnson Transformation

We then pass the pre-processed data $(P)_{N \times T}$ to the data transformation phase, where the Yeo-Johnson transformation is used [53]. The Yeo-Johnson transformation is better than other transformation methods because it produces well-organised data that is easier to use. Correctly formatted and transformed data improves the data quality and protects applications from potential landmines, such as null values, unexpected duplicates, incorrect indexing, and incompatible formats. Furthermore, the Yeo-Johnson transformation can make the data distribution more symmetric, and it does not require that the value be strictly positive. We estimate the parameters by applying this transformation to the response variable using the maximum penalised likelihood model. The Yeo-Johnson transformation expression is shown in Equation (8-3).

$$J = \beta(\delta, M) = \begin{cases} ((M + 1)^\delta - 1)/\delta; & M \geq 0, \delta \neq 0 \\ \log(M + 1); & M \geq 0, \delta = 0 \\ -\frac{((-M+1)^{2-\delta}-1)}{(2-\delta)}; & M < 0, \delta \neq 2 \\ -\log(-M + 1); & M < 0, \delta = 2 \end{cases} \quad (8-3)$$

Where $J = \beta(\delta, M)$ is the output of the data transformation of R_j , δ is any real number of the power parameter in a piecewise function form that makes it

continuous at the point of singularity ($\delta = 0$), where $\delta = 1$ is an identity transformation and M is the data vector with zero, negative and positive values or observations without restricting the type of observation needed stabilising the variance of the input datasets, which increase the features distribution symmetry and improving the validity of association measures (such as the correlation between features).

8.4.4 Feature Fusion using Proposed SSDCA-based Deep RNN

Once the data transformation is performed, we do the feature fusion based on the trained Deep RNN model of our SSDCA algorithm. In feature fusion, SSDs have greater throughput, continuous access times for quicker boot-ups, faster file transfers, and excellent performance. The CAViaR model specifies the evolution of the quantile over time using a special type of autoregressive process described in detail later. It applies to real data and can adapt to new risk environments. Thus, the SSDCA has the advantages of SSD and CAViaR in feature fusion. After data transformation, the size of the data remains $N \times T$; hence for reducing the features, the feature fusion step is fundamental in determining flight delays effectively. For example, the T attributes have six columns, meaning six features, as shown in the set $\{f_1, f_2, f_3, f_4, f_5, f_6\}$.

Correlation-based Feature Sorting

In this step, we perform the correlation for the six features. For example, we compute the correlation based on f_1 target values; thus, we get six correlation values. Then, we couple the features and change the columns based on high correlation values, which precedes the feature fusion process for reducing the features. We applied Spearman's rho correlation because our data has an ordinal level of measurement and follows a categorical distribution to measure the pairwise monotonicity of relationships by ranking from highest to lowest and choosing the high correlation values. Also, each variable changes in one direction of category without necessarily at the same rate. It can be calculated as shown in Equation (8-4).

$$C_s = 1 - \frac{6 \sum d_i^2}{(n^2 - n)} \quad (8-4)$$

Where C_s is the rank between two features, d_i is the difference between two variable ranks for each data pair while $\sum d_i^2$ Represents the square sum difference between two features' rank, and n is the number of instances.

Feature Fusion and Determination of β based on Deep Recurrent Neural Network

After feature fusion, we determine the β based on the Deep RNN classifier. Here, we take the features as the Deep RNN classifier's input for performing feature fusion using weights and bias-related with hidden layers.

a) Feature fusion

Once we change the columns based on the correlation values, we perform the feature fusion based on Equation (8-5).

$$f_k^{fused} = \sum_{i=1}^S \frac{\beta}{j} f_k \quad (8-5)$$

where $i = 1 + \frac{T}{l}$, $j = 1 \dots l$, T represents the total features and l refers to the number of selected features as shown in Equation (8-6).

$$l = \frac{T}{S}; 1 \leq k \leq S \quad (8-6)$$

where the term k is the index of fused features and S is the last index of the fused features.

b) Structure of Deep RNN

This network architecture contains several recurrent hidden layers in the network hierarchy. However, the recurrent connection exists only among hidden layers. It takes the previous state output as input to the next state, and the iteration process begins with the hidden state information. The advantage of using a Deep RNN [54] classifier over state-of-art classifiers is that it operates effectively under changes in the input feature length. A recurrent feature associated with Deep RNN yields high performance in feature fusion accuracy and the required number of iterations. We used Deep RNN to find the best parameter values because it can process inputs of any length and approximate any function. We illustrate the structure of the Deep RNN classifier in Figure (8-3).

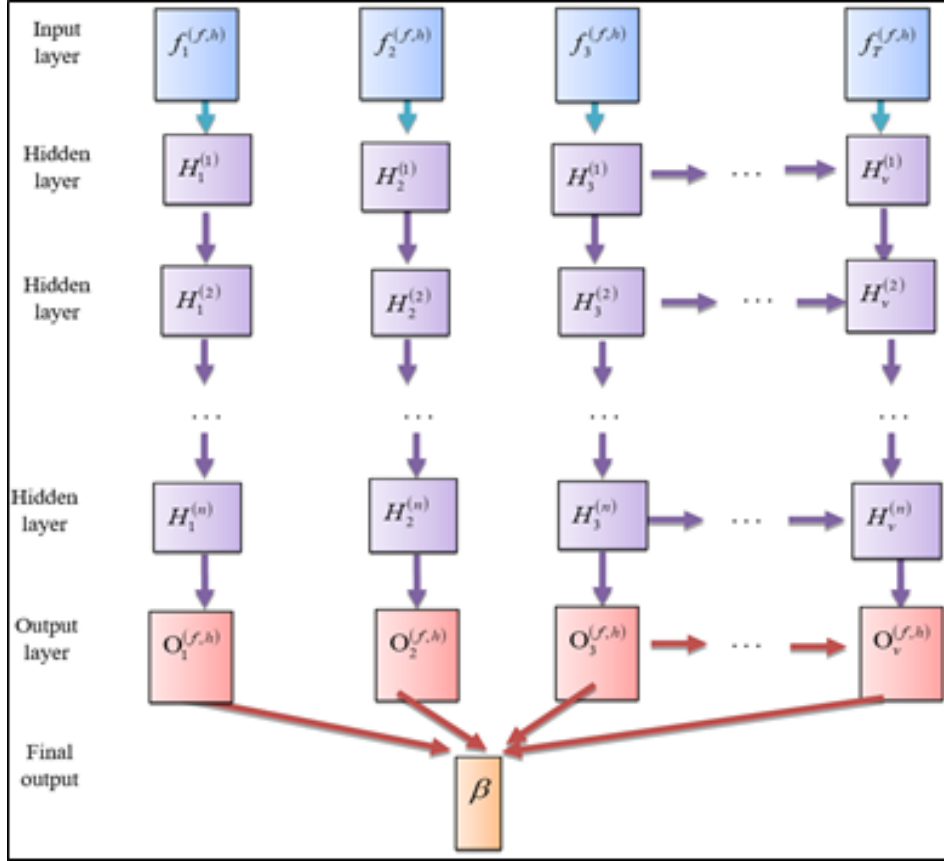


Figure 8-3 Structure of Deep RNN

The Deep RNN classifier defined the input vector of the layer f at time h as $U^{(f,h)} = \{U_1^{(f,h)}, U_2^{(f,h)}, \dots, U_r^{(f,h)}, \dots, U_n^{(f,h)}\}$ and the output vector of the layer f at time h as $O^{(f,h)} = \{O_1^{(f,h)}, O_2^{(f,h)}, \dots, O_r^{(f,h)}, \dots, O_s^{(f,h)}\}$. The set of units for each element of output and input vectors n refers to the arbitrary unit number of layers f and v signifies total units. However, the total units of $(f-1)^{th}$ layer and the arbitrary unit number γ and φ . Here, the input propagation weight from $(f-1)^{th}$ layer to f^{th} layer is shown by $W^{(f)} \in I^{v \times \gamma}$, and f^{th} we denote recurrent layer weight as $w^{(f)} \in I^{v \times v}$ the term I signifies weight set. Therefore, the input vector components are as in Equation (8-7).

$$U_n^{(f,h)} = \sum_{u=1}^{\gamma} z_{nu}^{(f)} O_z^{(f-1,h)} + \sum_{n'}^v \varepsilon_{nn'}^{(f)} O_{n'}^{(f,h-1)} \quad (8-7)$$

where, $z_{nn'}^{(f)}$ and $\varepsilon_{nn'}^{(f)}$ refer to elements of $W^{(f)}$, and $w^{(f)}$ the term n' signifies the arbitrary unit number of f^{th} layers. Thus, the output vector of f^{th} the layer elements is as in Equation (8-8).

$$O_n^{(f,h)} = \mu^{(f)}(U_n^{(f,h)}) \quad (8-8)$$

where $\mu^{(f)}$ refers to the activation function. Thus, activation functions, like the Sigmoid function, are indicated by $\mu(U) = \tanh(U)$ rectified linear unit function (ReLU) $\mu(U) = \max(U, 0)$, and we represent the logistic sigmoid function as $\mu(U) = \frac{1}{1+e^{-U}}$ the employed activation function. To simplify the prediction

process, consider ℓ^{th} weight as $z_{n\ell}^{(f)}$ and ℓ^{th} unit as $O_\ell^{(f-1,h)}$ and hence, we specified the bias as shown in Equation (8-9).

$$O^{(f,h)} \text{ or } \beta = \mu^{(f)}(W^{(f)}O^{(f-1,h)} + w^{(f)} \cdot O^{(f,h-1)}) \quad (8-9)$$

Thus, the output of the classifier is indicated as $O^{(f,h)}$ or β . And $W^{(f)}$ is the input propagation weight, recurrent layer weight as $w^{(f)}$, unit as $O_\ell^{(f-1,h)}$ and $\mu^{(f)}$ refers to the activation function. The classifier is tuned by a proposed optimisation algorithm for updating the classifier's weights, enabling effective prediction performance.

c) Training data

After the determination β , we computed the training data value by correlating the mean vector of data belonging to the class. The training data step is significant for the selected six features instance in model building to generate an output, as shown in Table 8-2.

Table 8-2 Determination of training data

d_i/f_T	f_1	f_2	f_3	f_T	Target
d_1	$C(d_1, f_1)$	$C(d_1, f_2)$	$C(d_1, f_3)$	$C(d_1, f_T)$	$\beta_{training}$
d_2	$C(d_2, f_1)$	$C(d_2, f_2)$	$C(d_2, f_3)$	$C(d_2, f_T)$	$\beta_{training}$
d_3	$C(d_3, f_1)$	$C(d_3, f_2)$	$C(d_3, f_3)$	$C(d_3, f_T)$	$\beta_{training}$
d_4	$C(d_4, f_1)$	$C(d_4, f_2)$	$C(d_4, f_3)$	$C(d_4, f_T)$	$\beta_{training}$

We compute the output training β using Equation (8-10).

$$\beta_{training}^{(i)} = \text{Correlation}(d_i, M_{d_i}) \quad (8-10)$$

where M_{d_i} refers to the mean vector of data d_i belonging to the class.

d) Training of Deep Recurrent Neural Network using Social Ski Driver Conditional Autoregressive

We carried out the training process of Deep RNN [54] using the developed SSDCA technique for finding optimal weights to tune the Deep RNN [24] for feature fusion and flight delay prediction. The feature fusion based on the developed SSDCA categorises data by obtaining optimal weights and dealing with new data characteristics from distributed resources. The naturally inspired SSD algorithm [55] has several evolutionary optimisation algorithms for reducing parameters and improving system performance. SSD aims to search in space for near-optimal or optimal solutions. Thus, this approach efficiently generates improved features for solving multi-aim optimisation problems. The method solves highly non-linear problems with complicated constraints and can deal with heterogeneous data.

On the other hand, the CAViaR model [59] has received much attention for distributing direct returns to the quantile behaviour. We employ regression quantile to estimate and update the parameters. Tests of the model adequacy use a criterion independent of each probability period of all past information processes. The method also increased the convergence process and the diversity of solutions and improved the balance between exploitation and exploration. CAViaR models can adapt to new risk environments. Thus, integrating CAViaR with SSD enhances the overall algorithmic performance. We give the algorithmic steps of the proposed SSDCA as follows.

i) Initialisation: The initial step of the SSDCA algorithm is the search agent's location initialisation, where the user determines the total number of agents. As shown in Equation (8-11), the agent's position is.

$$X_v^t; (1 \leq v \leq z) \quad (8-11)$$

where the term X_v^t refers to the agent's location at a time t , v is the velocity and z is the total number of samples.

ii) Objective function evaluation: we select the optimal agent location using the minimal learning error as an optimal solution. We estimate the objective function error using Equations (8-12), Equation (8-13), Equation (8-14) and Equation (8-15).

$$MSE = \frac{1}{z} \sum_{h=1}^z (I_h - \beta_h)^2 \quad (8-12)$$

$$RMSE = \sqrt{\frac{1}{z} \sum_{h=1}^z (I_h - \beta_h)^2} \quad (8-13)$$

$$MAE = \frac{1}{z} \sum_{h=1}^z |I_h - \beta_h| \quad (8-14)$$

$$AC = \frac{TRUE\ POSITIVE + TRUE\ NEGATIVE}{TRUE\ POSITIVE + FALSE\ POSITIVE + TRUE\ NEGATIVE + FALSE\ NEGATIVE} \quad (8-15)$$

where the classifiers estimated target, the output is I_h and β_h . The term z denotes the total number of samples.

iii) Solution update using the SSDCA algorithm: Once we compute the objective functions, the solution undergoes the location update based on SSDCA, as shown in Equation (8-16). The standard equation of the SSD velocity K_{lm}^{t+1} is given by Equation (8-17).

$$X_{lm}^{t+1} = X_{lm}^t + K_{lm}^{t+1} \quad (8-16)$$

$$K_{lm}^{t+1} = \begin{cases} e \sin(m_1)(B_{lm}^t - X_{lm}^t) + \sin(m_1)(A_{lm}^t - X_{lm}^t); s_2 \leq 0.5 \\ e \cos(m_1)(B_{lm}^t - X_{lm}^t) + \cos(m_1)(A_{lm}^t - X_{lm}^t); s_2 > 0.5 \end{cases} \quad (8-17)$$

where K_{lm} signifies velocity of X_{lm} , and uniformly distributed random numbers represent m_1 and m_2 range between 0 to 1. B_{lm} refers to the optimal agent solution and A_{lm} denotes means a global solution for the entire population. Hence, Equation (8-17) depicts the standard SSD equation incorporating the updated CAViaR equation. Therefore, the CAViaR standard equation is given as in Equation (8-18).

$$X_{lm}^t = \alpha_p + \sum_{h=1}^k \alpha_h X_{lm}(t-h) + \sum_{l=1}^r \alpha_l f(X_{lm}(t-h)) \quad (8-18)$$

Substituting $k = r = 2$ in Equation (8-18) becomes Equation (8-19).

$$X_{lm}^t = \alpha_0 + \alpha_1 X_{lm}(t-1) + \alpha_2 X_{lm}(t-2) + \alpha_1 f(X_{lm}(t-1)) + \alpha_2 f(X_{lm}(t-2)) \quad (8-19)$$

where $f(\cdot)$ denotes the fitness function considering the case-1 in Equation (8-16), which we refer to as Equation (8-16a).

Now substituting Equation (8-17a) in Equation (8-16) results to:

$$X_{lm}^{t+1} = X_{lm}^t + e \sin(m_1)(B_{lm}^t - X_{lm}^t) + \sin(m_1)(A_{lm}^t - X_{lm}^t) \quad (8-20)$$

Rearranging, Equation (8-20), the solution becomes,

$$X_{lm}^{t+1} = X_{lm}^t [1 - e \sin(m_1) - \sin(m_1)] + e \sin(m_1) B_{lm}^t + \sin(m_1) A_{lm}^t \quad (8-21)$$

Substituting Equation (8-19) in Equation (8-21) yields:

$$X_{lm}^{t+1} = \alpha_0 + \alpha_1 X_{lm}(t-1) + \alpha_2 X_{lm}(t-2) + \alpha_1 f(X_{lm}(t-1)) + \alpha_2 f(X_{lm}(t-2)) \\ [1 - e \sin(m_1) - \sin(m_1)] + e \sin(m_1) B_{lm}^t + \sin(m_1) A_{lm}^t \quad (8-22)$$

Following the same procedure when considering case-2 in Equation (8-17), which we refer to as Equation (8-17b), we get:

$$K_{lm}^{t+1} = X_{lm}^t [e \cos(m_1) + \cos(m_1)] + e \cos(m_1) B_{lm}^t + \cos(m_1) * A_{lm}^t \quad (8-23)$$

Similarly, substituting Equation (8-19) in Equation (8-23) results to:

$$K_{lm}^{t+1} = \alpha_0 + \alpha_1 X_{lm}(t-1) + \alpha_2 X_{lm}(t-2) + \alpha_1 f(X_{lm}(t-1)) + \alpha_2 f(X_{lm}(t-2)) \\ [e \cos(m_1) + \cos(m_1)] + e \cos(m_1) B_{lm}^t + \cos(m_1) * A_{lm}^t \quad (8-24)$$

Thus, the final expression for the updated equation of the proposed SSDCA applied in performing the flight delay prediction is shown below in Equation (8-25):

$$K_{lm}^{t+1} = \begin{cases} \alpha_0 + \alpha_1 X_{lm}(t-1) + \alpha_2 X_{lm}(t-2) + \alpha_1 f(X_{lm}(t-1)) + \alpha_2 f(X_{lm}(t-2)) \\ [1 - e \sin(m_1) - \sin(m_1)] + e \sin(m_1) B_{lm}^t + \sin(m_1) A_{lm}^t & ; s_2 \leq 0.5 \\ \alpha_0 + \alpha_1 X_{lm}(t-1) + \alpha_2 X_{lm}(t-2) + \alpha_1 f(X_{lm}(t-1)) + \alpha_2 f(X_{lm}(t-2)) \\ [e \cos(m_1) + \cos(m_1)] + e \cos(m_1) B_{lm}^t + \cos(m_1) * A_{lm}^t & ; s_2 > 0.5 \end{cases} \quad (8-25)$$

iv) Recheck the feasibility: Once we test the updated position and each solution of the objective functions, we consider the optimal solution to be the one with maximal fitness.

v) Termination: The steps *i* to *iv* continue until we meet a specified maximum iteration number or achieve an optimal solution. The pseudocode of the developed SSDCA approach is shown in Algorithm (8-1).

Algorithm 8-1: Pseudocode of the SSDCA-based Deep RNN algorithm

```

1:  Input: Search agents location  $X_v^t; (1 \leq v \leq z)$ 
2:  Output: Optimal location
3:  Procedure:
4:  begin
5:      Agent's velocities and position are initialised (8-11)
6:      Consider fitness value should be minimal
7:      while stopping criteria are not met, do
8:          for whole agents do
9:              Compute fitness value based on equation (8-12)
10:             Sort agents based on fitness value
11:             Determine mean global solution and previous
                optimal solution
12:             Update the SSD (8-16) using (8-17)
13:             Update the CAViaR (8-18) using (8-19)
14:             Update the SSDCA (8-24) based on (8-25)
15:          end for
16:      Check solution feasibility
17:      Return optimal solution
18:       $t = t + 1$ 
19:  end while
20:  The best solution is achieved
21: End

```

We call F the output obtained from line 20 of Algorithm (8-1).

Flight delay prediction based on proposed SSDCA-based Deep LSTM

The feature fusion output F is plugged into the Deep LSTM classifier [57] for predicting the flight delay training by the novel SSDCA, incorporating the SSD algorithm [55] and CAViaR [56]. We employ the Deep LSTM classifier because processing the input time series data takes less computational time. Another significant benefit of Deep LSTM is that it requires less training data and can generate optimal results at a specific instance. This section elaborates on the Deep LSTM structure and framework utilised for optimising training weights.

8.4.5 Deep LSTM Architecture

We feed the Deep LSTM classifier with the fusion features F obtained from the feature fusion to accomplish the flight delay prediction. The Deep LSTM effectively achieves flight delay prediction by applying the classifier's memory cell, which is advantageous for the other classifiers. Here, the memory cell utilises the stored state information and acts as an accumulator. The Deep LSTM uses the input and their neighbours' past states to predict future states based on the convolutional operator and Hadamard product. We achieve an effective flight delay prediction using the high transitional kernel where the encoding and forecasting layers form the Deep LSTM structure. The forecasting network receives the initial input and outputs the cell encoding network. Also, the Deep LSTM uses a self-parametrised gate to clear cell access. The memory cell receives the state information if the gate gets activated when subject to the input. The classifier forgets the past cell information whenever the forget gate is ON. Because of insufficient decaying error backflow, information storage takes a long over extended intervals with recurrent backpropagation. A significant advantage of the classifier is how it manages the information flow.

LSTM is an efficient and gradient-based method. The LSTM can bridge the minimal time lags learning to enforce constant error flow through "constant error carousels" over 1000 discrete time steps within the particular unit. In terms of space and time, the LSTM is local. The LSTM algorithm can solve long-time lags of complex tasks that other recurrent neural network algorithms cannot. Here, a convolutional architecture exists in the input to the state-to-state transition, so the forecasting problem solves a structure in the LSTM. The classifier's rows and

columns are the 3D tensors in spatial dimensions with inputs, hidden states, and gates utilised. We define the input cell and states as vectors in spatial grid form to enhance the flight delay prediction performance. However, the classifier employs the previous state and neighbours' input cells to compute future states. Thus, the Deep LSTM classifier includes an input $\{T_1, \dots, T_m\}$, hidden states $\{I_1, \dots, I_m\}$, cell output $\{D_1, \dots, D_m\}$, and gates x_m, y_m, z_m . Figure 8-4 describes the architecture of the Deep LSTM classifier.

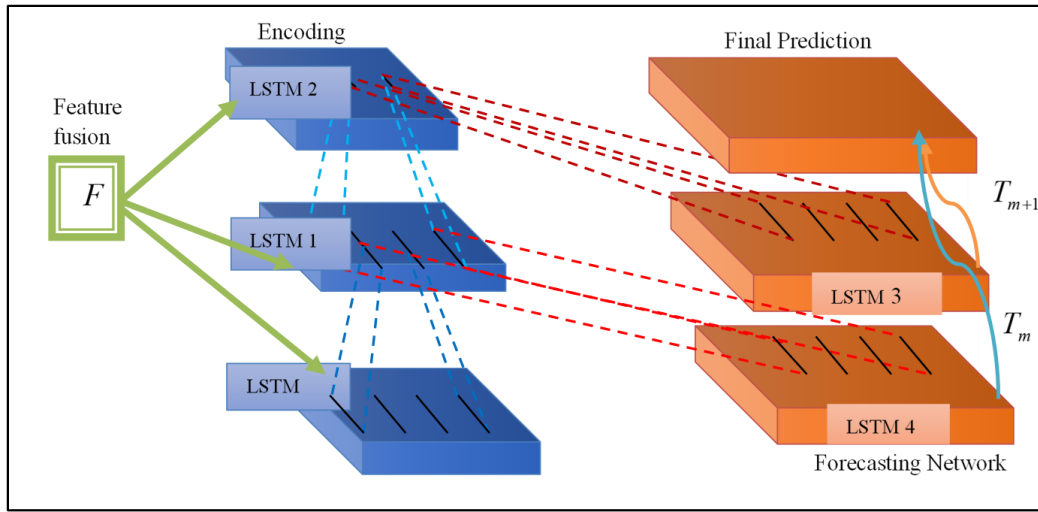


Figure 8-4 Architecture of Deep LSTM Classifier

Equation (8-26) defines the input gate output:

$$x_m = \gamma(S_x^T * T_m + S_x^I * I_{m-1} + S_x^D \circ D_{m-1} + \lambda^x) \quad (8-26)$$

Where T_m is the input vector and S_x^T is the weight among the input layer and gate. The gate activation function is given by γ , and S_x^D is the weight vector between the cell input and output layer, and S_x^I is the weight between memory input and output layer, while I_{m-1} represents the previous cell output, and D_{m-1} is the previous memory unit output. The term λ^x represents input layer bias, the character $*$ represents the Convolutional operator, and the character $\circ$ is the element-wise multiplication. Equation (8-27) represents the forget gate output:

$$y_m = \gamma(S_y^T * T_m + S_y^I * I_{m-1} + S_y^D \circ D_{m-1} + \lambda^y) \quad (8-27)$$

where S_y^T denotes weight among the input layer forget gate while the term S_y^I is the weight among the memory unit of the previous layer and output gate and S_y^D is the weight among output gate and cells. The term λ^y represents the bias that relates to forgetting gates. Equation (8-28) represents the output generated from the output gate:

$$z_m = \gamma(S_z^T * T_m + S_z^I * I_{m-1} + S_z^D \circ D_{m-1} + \lambda^z) \quad (8-28)$$

where S_z^T represents the weight between the input layer and the output gate, the weight between the output gate and memory unit, and the weight between the output gate and the cell; the term λ^z is the output gate bias based on the activation function. Equation (8-29) represents the temporary cell state output.

$$\tilde{D}_m = \tanh(S_v^T * T_m + S_v^I * I_{m-1} + \lambda^v) \quad (8-29)$$

where λ^v represents the bias and S_v^T the weight between the cell and the input layer, while the symbol represents the weight between the cell and the memory unit. The cell output estimates adding the previous and current layer, temporary cell state and a memory unit. Equation (8-30) and Equation (8-31) represent the cell output estimation.

$$D_\tau = y_m \circ D_{m-1} + x_m \circ \tilde{D}_\tau \quad (8-30)$$

$$D_m = y_m \circ D_{m-1} + x_m \circ \tanh(S_v^T * T_m + S_v^I * I_{m-1} + \lambda^v) \quad (8-31)$$

Equation (8-32) expresses the memory unit generated output:

$$I_m = z_m \circ \tanh(D_m) \quad (8-32)$$

where z_m represents the output gate, and the term I_m represents the block output memory. Thus, Equation (8-33) represents the generated output O_m :

$$O_m = \alpha(S_o^I \cdot I_m + \lambda^o) \quad (8-33)$$

where the output vector of weight among the memory unit is s_o^i and λ^o represents the output layer bias.

8.5 Results and Discussion

This section describes the results of the proposed SSDCA-based Deep LSTM based on some benchmark metrics and compares our method with a set of methods from the literature. Table 8-3 presents the considered methods and the corresponding special names we give them.

Table 8-3 Methods

S/No.	Methods	Special Name
1	Deep Belief Network (DBN)	Med 1
2	Gradient boosting classifier	Med 2
3	Information gain-SVM	Med 3
4	Multi-agent approach	Med 4
5	Deep LSTM	Med 5
6	PSO+Deep LSTM	Med 6
7	ACO+Deep LSTM	Med 7
8	SSD+Deep LSTM	Med 8
9	HBO+Deep LSTM	Med 9
10	EWA+Deep LSTM	Med 10

8.5.1 Dataset Source

The dataset considered for the experimentation is the flight delay prediction dataset with US flight data downloaded from the US Government Bureau of transportation statistics [61, 62].

January Flight Delay Prediction dataset (dataset-1 [61]): This dataset contains the flights in January 2019 and January 2020. It contains over 400,000 flights, which translates into 400,000 rows. It has 21 feature columns that specify the features of each flight. It includes the destination airport, origin airport, departure time, arrival time, and aircraft information. We used the dataset for predicting flight delays at the destination airport for January.

February Flight Delay Prediction dataset (dataset-2 [62]): This dataset contains the flights in February 2019 and February 2020. It contains over 400,000 flights, which translates into 400,000 rows. It has 21 feature columns that specify

the features of each flight. It includes the destination airport, origin airport, departure time, arrival time, and aircraft information. We used the dataset for predicting flight delays at the destination airport for February.

8.5.2 Evaluation Metrics

We evaluate the proposed model performance using the MSE, RMSE, MAE and Accuracy metrics [63, 64, 65, 66].

MSE measures the average square difference between the estimated and target values, as shown in Equation (8-10).

RMSE measures the square root of the average square difference between the actual and the predicted value, as shown in Equation (8-11).

MAE: This measures the Mean of the absolute values of individual prediction errors over all instances in the test set, as shown in Equation (8-12).

Accuracy: This measures the estimated value's closeness to a standard or actual value, as shown in Equation (8-13).

8.5.3 Method Comparisons

We analyse the developed model performance and compare it with the existing methods from the literature [1, 3, 5, 23] and other meta-heuristic approaches implemented with the Deep LSTM in our study.

Model Evaluation

In this section, we compare the proposed SSDCA-based Deep LSTM by varying the training and testing datasets percentage to investigate the impact on the model's performance.

Analysis using dataset-1

Figure 8-5 in the appendix illustrates the analysis of the developed approach based on dataset-1 by considering the feature size as '6'. Figure (8-5a) shows the analysis of RMSE regarding training data, with smaller values of RMSE being the favourable ones. When we consider 60% as training data, RMSE measured shows that our proposed method outperforms the existing methods, as shown in Table 8-4. For 90% training data, the developed model gained a minimum RMSE of 0.1114, whereas the remaining existing methods did not perform as well, with the actual values shown in Table 8-4.

Figure 8-5b shows the analysis of MSE in terms of training and testing data percentage. For 70% of training data, the MSE was measured by the existing

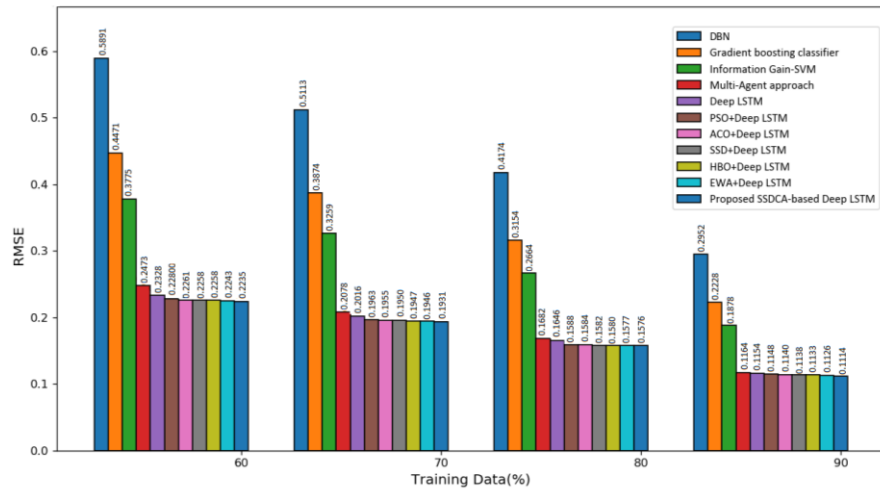
methods, and our proposed method is shown in Table 8-4. For 90% training data, the developed model achieved a minimum MSE of 0.0134 compared to the other methods, with detailed values shown in Table 8-4.

Figure 8-5 shows the analysis of MAE in terms of training and testing data percentage. For 60% of training data, the MAE measured by our proposed method outperforms the other existing methods with a value of 0.1615. Detailed values are shown in Table 8-4. For 90% training data, the developed model achieved a minimum MAE of 0.0511 compared to the other methods, with detailed values shown in Table 8-4.

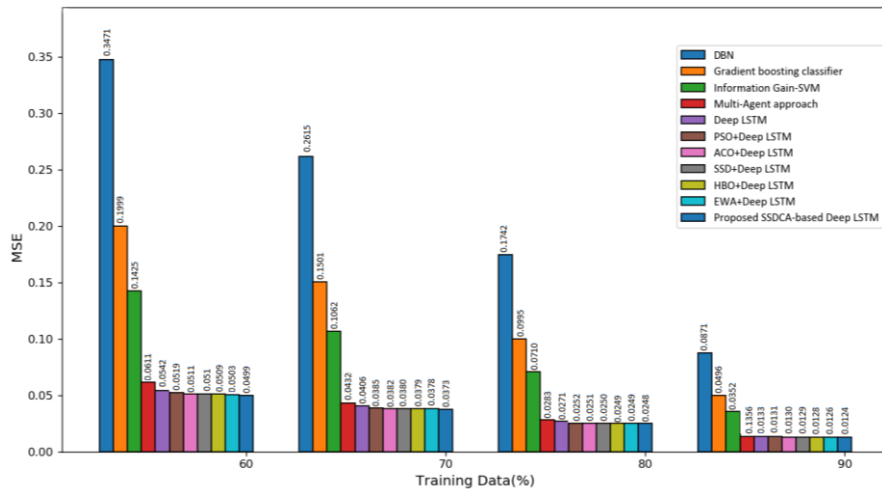
Figure (8-5d) shows the analysis of Accuracy in terms of training and testing data percentage. For 70% of training data, the Accuracy measured by our proposed method outperforms the other existing methods with a value of 0.9156. Table 8-4 shows the detailed values of the other existing methods. For 90% of training data, the developed model achieves an accuracy of 0.9362 compared to the other methods, with detailed values shown in Table 8-4.

Table 8-4 Results comparison of our proposed method with other existing methods across different evaluation metrics on dataset-1

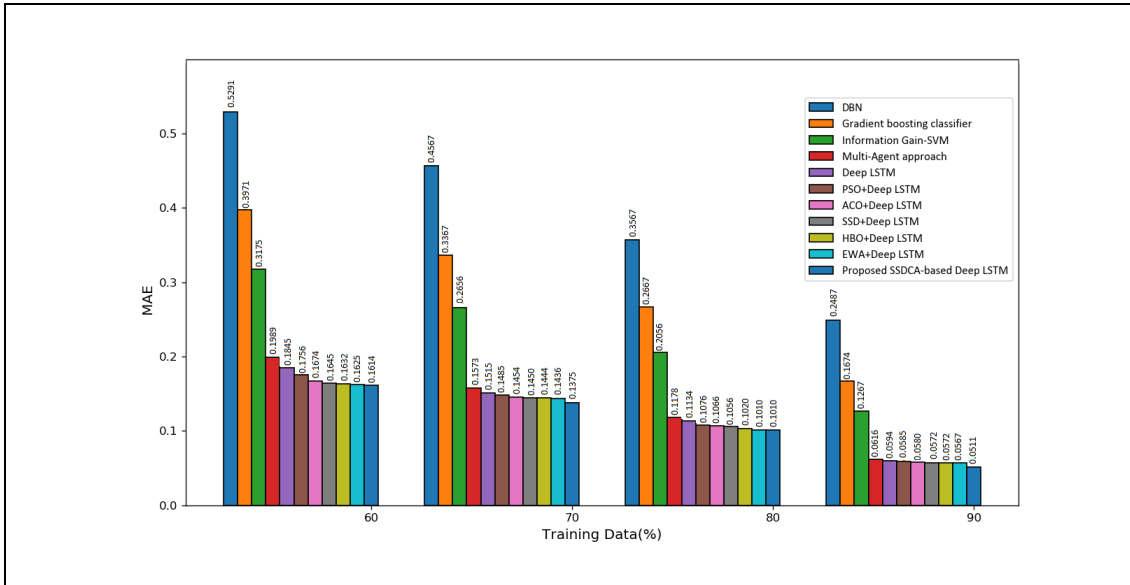
S/ No	Training/ Testing (%)	Method/ Metrics	Med 1	Med 2	Med 3	Med 4	Med 5	Med 6	Med 7	Med 8	Med 9	Med 10	Proposed SSDCA- based Deep LSTM
1	60/40	RMSE	0.589 1	0.44 71	0.37 75	0.24 73	0.23 28	0.22 80	0.22 61	0.22 58	0.22 58	0.22 43	0.2235
		MSE	0.347 1	0.19 99	0.14 25	0.06 11	0.05 42	0.05 19	0.05 11	0.05 11	0.05 09	0.05 03	0.0499
		MAE	0.529 1	0.39 71	0.31 75	0.19 89	0.18 45	0.17 56	0.16 74	0.16 45	0.16 32	0.16 25	0.1614
		Accuracy	0.752 8	0.76 56	0.78 57	0.79 24	0.80 67	0.81 56	0.82 37	0.83 98	0.84 78	0.87 56	0.9013
2	70/30	RMSE	0.521 3	0.38 74	0.32 59	0.20 78	0.20 16	0.19 63	0.19 55	0.19 50	0.19 47	0.19 46	0.1931
		MSE	0.261 5	0.15 01	0.10 62	0.04 32	0.04 06	0.03 85	0.03 82	0.03 80	0.03 79	0.03 78	0.0373
		MAE	0.456 7	0.33 67	0.26 56	0.15 73	0.15 15	0.14 85	0.14 54	0.14 50	0.14 44	0.14 36	0.1375
		Accuracy	0.759 8	0.77 20	0.78 96	0.79 95	0.81 23	0.81 95	0.82 85	0.84 25	0.85 34	0.87 98	0.9156
3	80/20	RMSE	0.417 4	0.31 54	0.26 64	0.16 82	0.16 46	0.15 88	0.15 84	0.15 82	0.15 80	0.15 77	0.1576
		MSE	0.174 2	0.09 95	0.07 10	0.02 83	0.02 71	0.02 52	0.02 51	0.25 0	0.02 49	0.02 49	0.0248
		MAE	0.356 7	0.26 67	0.20 56	0.11 78	0.11 34	0.10 76	0.10 66	0.10 56	0.10 20	0.10 10	0.1010
		Accuracy	0.775 4	0.78 45	0.79 45	0.80 45	0.81 96	0.82 95	0.83 67	0.85 43	0.86 34	0.89 56	0.9268
4	90/10	RMSE	0.295 2	0.22 28	0.18 78	0.11 64	0.11 54	0.11 48	0.11 40	0.11 38	0.11 33	0.11 26	0.1114
		MSE	0.087 1	0.04 96	0.03 52	0.13 56	0.01 33	0.01 31	0.01 30	0.01 29	0.01 28	0.01 26	0.1024
		MAE	0.248 7	0.16 74	0.12 67	0.06 16	0.05 94	0.05 85	0.05 80	0.05 72	0.05 72	0.05 67	0.0511
		Accuracy	0.803 4	0.82 78	0.85 84	0.86 45	0.87 56	0.88 34	0.89 07	0.90 56	0.91 56	0.92 65	0.9361



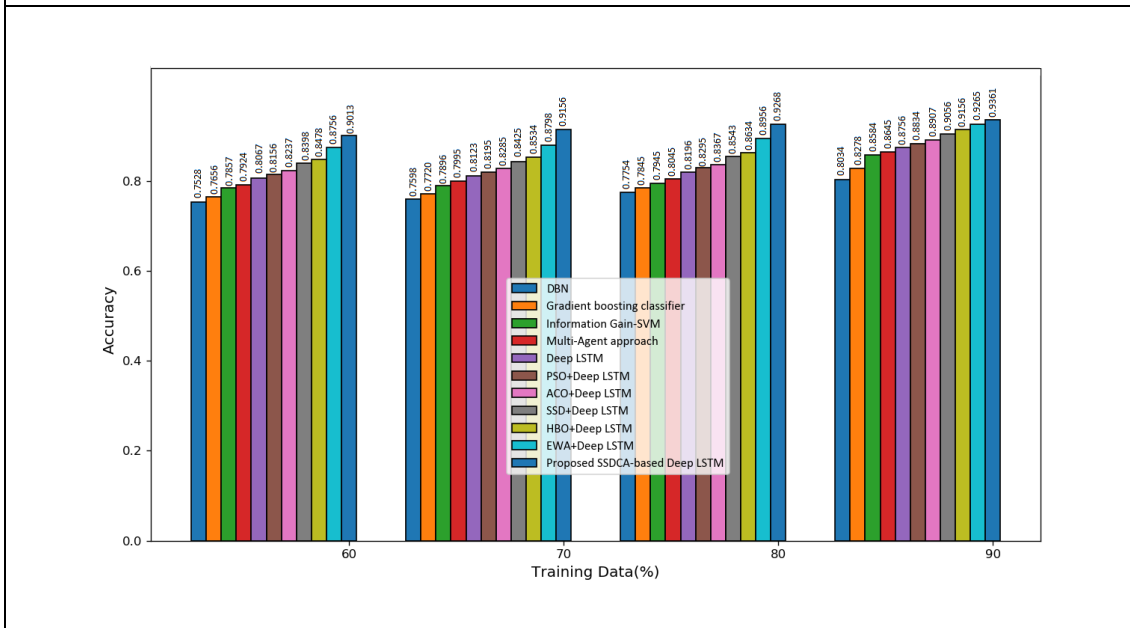
(a)



(b)



(c)



(d)

Figure 8-5 Analysis using Dataset-1 with feature size 6, a) RMSE, b) MSE, c) MAE, d) Accuracy

Analysis using dataset-2

Figure 8-6 in the appendix illustrates the analysis of the developed approach based on dataset-2 by considering the feature size as '6'. Figure 8-6a shows the analysis of RMSE regarding training data, with smaller values of RMSE being the favourable ones. When we consider 60% as training data, RMSE measured shows that our proposed method outperforms the existing methods, as shown in

Table 8-5. For 90% training data, the developed model gained a minimum RMSE of 0.1157, whereas the remaining existing methods did not perform as well, with the actual values shown in Table 8-5.

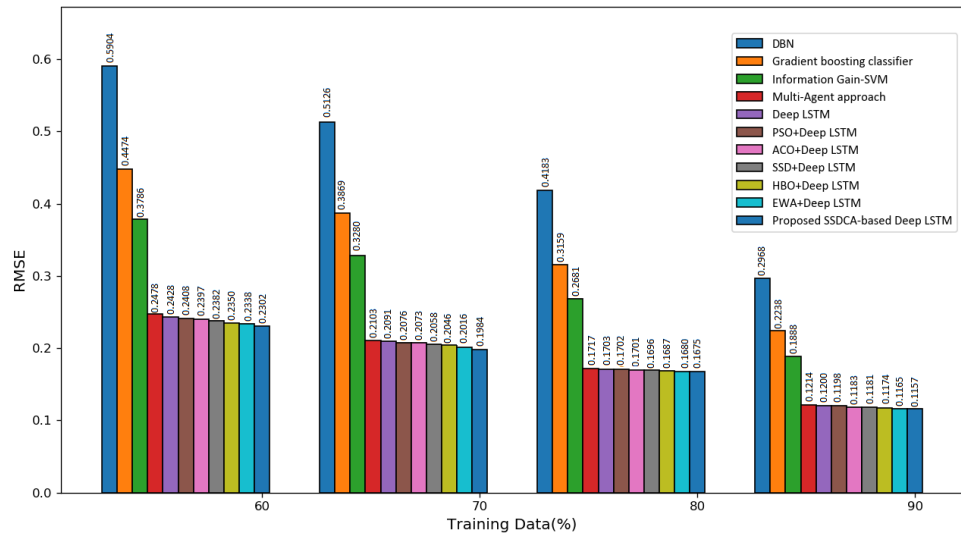
Figure 8-6b shows the analysis of MSE in terms of training and testing data percentage. For 70% of training data, the MSE was measured by the existing methods, and our proposed method is shown in Table 8-4. For 90% training data, the developed model achieved a minimum MSE of 0.0134 compared to the other methods, with detailed values shown in Table 8-5.

Figure 8-6c shows the analysis of MAE in terms of training and testing data percentage. For 60% of training data, the MAE measured by our proposed method outperforms the other existing methods with a value of 0.1701. Detailed values are shown in Table 8-5. For 90% training data, the developed model achieved a minimum MAE of 0.0557 compared to the other methods, with detailed values shown in Table 8-5.

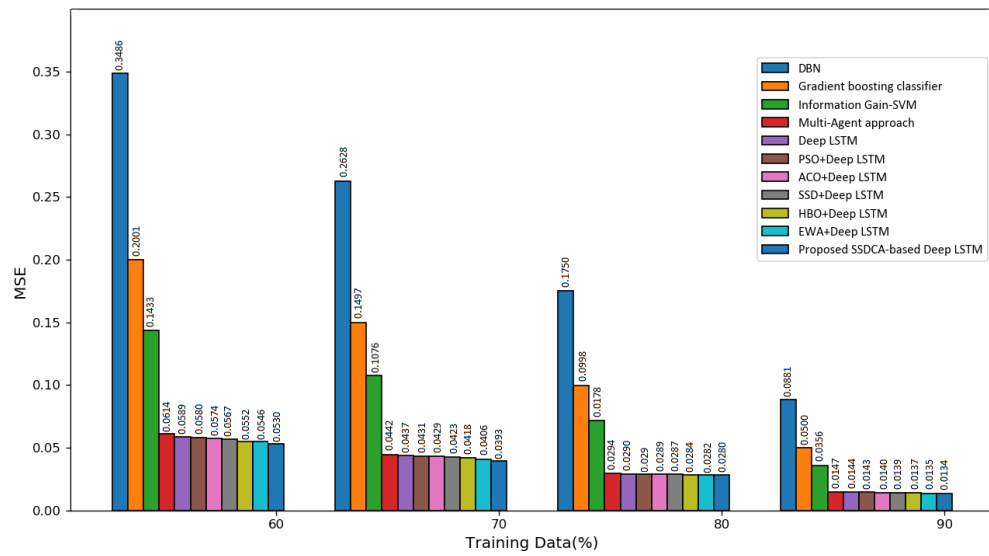
Figure 8-6d shows the analysis of Accuracy in terms of training and testing data percentage. For 70% of training data, the Accuracy measured by our proposed method outperforms the other existing methods with a value of 0.9085. Table 8-5 shows the detailed values of the other existing methods. For 90% of training data, the developed model achieves an accuracy of 0.9252 compared to the other methods, with detailed values shown in Table 8-5.

Table 8-5 Results comparison of our proposed method with other existing methods across different evaluation metrics on dataset-2

S/ No	Training/ Testing (%)	Method/ Metrics	Med 1	Med 2	Med 3	Med 4	Med 5	Med 6	Med 7	Med 8	Med 9	Med 10	Propose d SSDCA- based Deep LSTM
1	60/40	RMSE	0.5904	0.4474	0.3786	0.2478	0.2428	0.2408	0.2397	0.2382	0.2350	0.2338	0.2302
		MSE	0.386	0.2001	0.1433	0.0614	0.0589	0.0580	0.0574	0.0567	0.0552	0.0546	0.0530
		MAE	0.5315	0.3956	0.3267	0.1967	0.1915	0.1895	0.1825	0.1795	0.1750	0.1732	0.1701
		Accuracy	0.7598	0.7756	0.7834	0.7956	0.8052	0.8167	0.8256	0.8367	0.8578	0.8626	0.8942
		RMSE	0.5126	0.3869	0.3280	0.2103	0.2091	0.2076	0.2073	0.2058	0.2046	0.2016	0.1984
2	70/30	MSE	0.2628	0.1497	0.1076	0.0442	0.0437	0.0431	0.0429	0.0423	0.0418	0.0406	0.0393
		MAE	0.4626	0.3367	0.2756	0.1576	0.1491	0.1476	0.1467	0.1445	0.1427	0.1406	0.1375
		Accuracy	0.7645	0.7834	0.7945	0.8013	0.8146	0.8245	0.8356	0.8467	0.8596	0.8694	0.9085
		RMSE	0.4183	0.3159	0.2681	0.1717	0.1703	0.1702	0.1701	0.1696	0.1687	0.1680	0.1675
		MSE	0.1750	0.0998	0.0178	0.0294	0.0290	0.0290	0.0289	0.0287	0.0284	0.0282	0.0280
3	80/20	MAE	0.3567	0.2658	0.2181	0.1217	0.1203	0.1202	0.1201	0.1196	0.1187	0.1180	0.1175
		Accuracy	0.7756	0.7956	0.8056	0.8145	0.8256	0.8352	0.8426	0.8567	0.8631	0.8745	0.9156
		RMSE	0.2968	0.2238	0.1888	0.1214	0.1200	0.1198	0.1183	0.1181	0.1174	0.1165	0.1157
		MSE	0.0881	0.0500	0.0356	0.0147	0.0144	0.0143	0.0140	0.0139	0.0137	0.0135	0.0134
		MAE	0.2368	0.2238	0.1388	0.0614	0.0600	0.0593	0.0583	0.0581	0.0573	0.0565	0.0557
4	90/10	Accuracy	0.7967	0.8042	0.8152	0.8221	0.8372	0.8442	0.8517	0.8615	0.8742	0.9042	0.9252



(a)



(b)

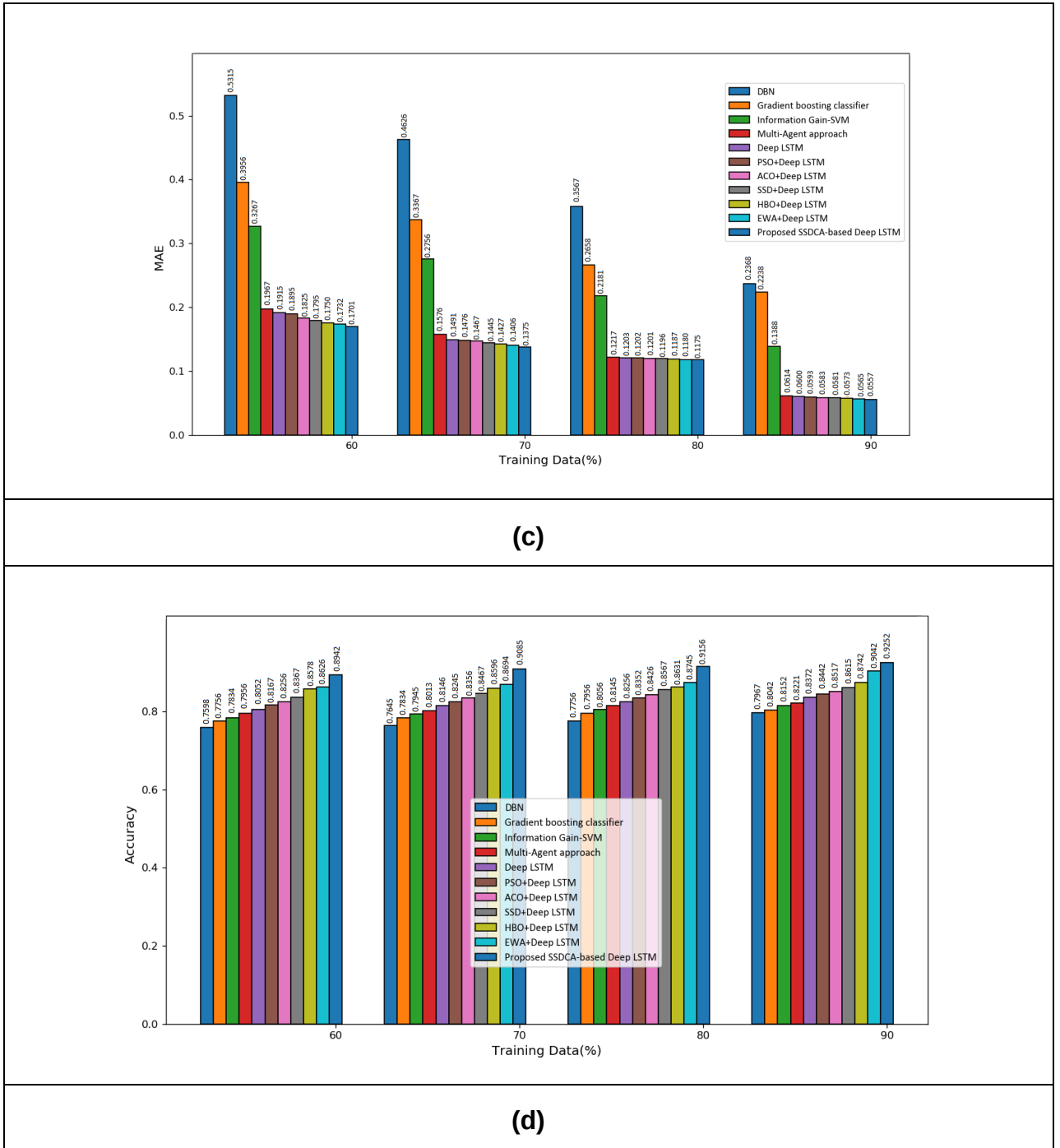


Figure 8-6 Analysis using Dataset-2 with feature size 6, a) RMSE, b) MSE, c) MAE, d) Accuracy

Delay prediction analysis

We now set the training data to be 70% of the dataset. Figure (8-7) represents the delay prediction gained using dataset-1. On January 15, 2020, the actual number of flights delayed was 1707, shown under the (original label). Our

proposed method predicts 1640 and is the most accurate method. Detailed results of all methods appear in Table 8-6.

Table 8-6 Delay prediction using dataset-1

Method	DBN	gradient boosting classifier	information gain-SVM	multi- agent approach	Deep LSTM	Proposed SSDCA- based Deep LSTM
Number of flights delayed	5300	3337	2534	1365	1365	1640

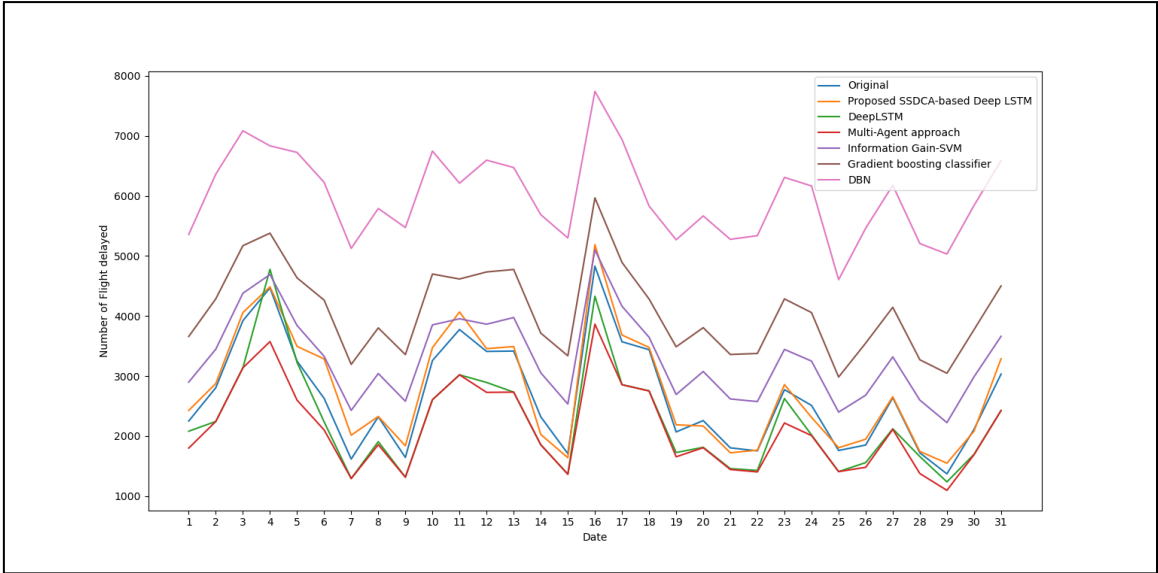


Figure 8-7 Delay Prediction using Dataset-1

We now set the training data to be 70% of the dataset. Figure 8-8 represents the delay prediction gained using dataset-1. On February 20, 2020, the actual number of flights delayed was 2772, shown under the (original label). Our proposed method predicts 2874 and is the most accurate method. Detailed results of all methods appear in Table 8-7.

Table 8-7 Delay prediction using dataset-2

Method	DBN	gradient boosting classifier	information gain-SVM	multi- agent approach	Deep LSTM	Proposed SSDCA- based Deep LSTM
Number of flights delayed	6472	4357	3525	2217	2217	2874

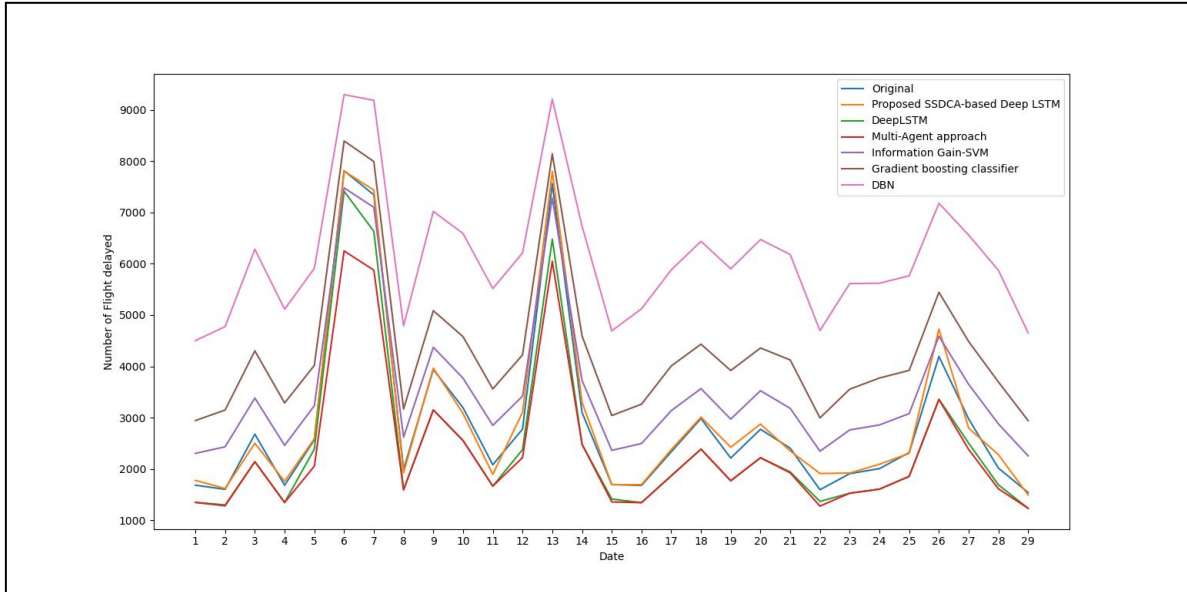
**Figure 8-8 Delay Prediction using Dataset-2****Convergence analysis**

Figure 8-9 shows the analysis for convergence using dataset-1. For dataset-1, the MSE of the considered Deep LSTM method coupled with the different optimisers. PSO+Deep LSTM is 0.0132, ACO+Deep LSTM is 0.0130, SSD+Deep LSTM is 0.0130, HBO+Deep LSTM is 0.0128, and EWA+Deep LSTM the convergence is 0.0127. The MSE of the proposed SSDCA-based Deep LSTM is 0.0124. Hence, the proposed algorithm has the best convergence compared to the other algorithms.

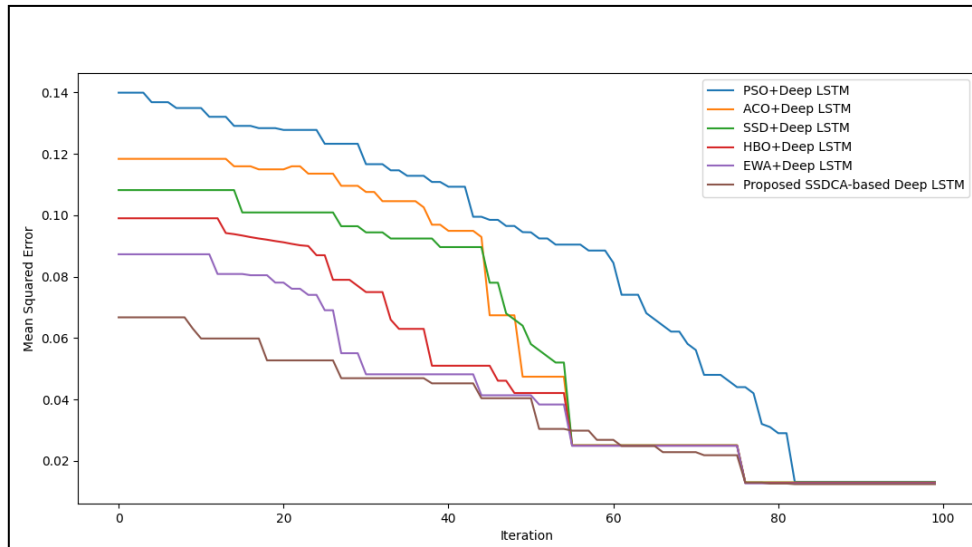


Figure 8-9 Convergence Analysis using Dataset-1

Figure 8-10 shows the analysis of convergence using dataset-2. For dataset-2, the MSE of the considered Deep LSTM method coupled with the different optimisers. PSO+Deep LSTM is 0.0157, ACO+Deep LSTM is 0.0148, SSD+Deep LSTM is 0.0136, HBO+Deep LSTM is 0.0130, and the convergence of EWA+Deep LSTM is 0.0119, and for the Proposed SSDCA-based Deep LSTM the convergence is 0.0108. Hence, the proposed algorithm has the best convergence compared to the other algorithms.

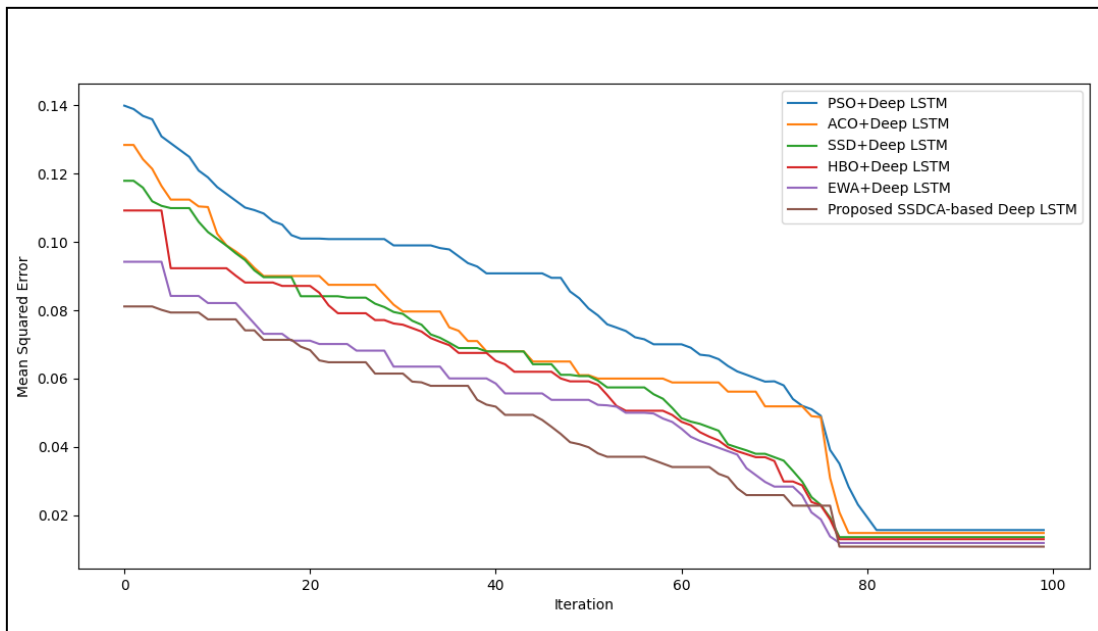


Figure 10: Convergence Analysis using Dataset-2

8.5.4 Comparative Discussion

Table 8-8 illustrates the comparative results of the proposed SSDCA-based Deep LSTM approach. When considering the feature size as '6' for dataset-1our proposed approach gained minimum RMSE and MSE values. The RMSE measured by the proposed SSDCA-based Deep LSTM is 0.1114. In contrast, the existing DBN, gradient boosting classifier, information gain-SVM, multi-agent approach, and Deep LSTM achieved the RMSE of 0.2953, 0.2229, 0.1878, 0.1165, and 0.1154, respectively. The MSE achieved by the existing DBN, gradient boosting classifier, information gain-SVM, multi-agent approach, Deep LSTM, and proposed SSDCA-based Deep LSTM for feature size '6' is 0.0872, 0.0497, 0.0353, 0.0136, 0.0133, and 0.0124 respectively. Table 2 shows that the proposed approach achieved minimal RMSE and MSE of 0.1065 and 0.0113 with dataset-2. The existing DBN, gradient boosting classifier, information gain-SVM, multi-agent approach, and Deep LSTM achieved the RMSE of 0.2971, 0.2240, 0.1892, 0.121, 0.1143 and 0.1065, respectively. The MSE achieved by the existing DBN, gradient boosting classifier, information gain-SVM, multi-agent approach, Deep LSTM, and proposed SSDCA-based Deep LSTM for feature size '6' is 0.0883, 0.0502, 0.0358, 0.0148, 0.0131 and 0.0113 respectively.

Our analysis shows that our proposed method offers higher accuracy and needs fewer iterations than the existing methods. Also, it is more versatile in terms of skewness. CAViaR models can adapt to new risk environments, and LSTMs avoid the long-term dependency problem. RNN can process inputs of any length, and RNN with a deep network can approximate any function. Deep RNN can be much more efficient regarding computation and the number of parameters.

Table 8-8 Comparative discussion

Dataset	Metrics	DBN	Gradient boosting classifier	Information gain-SVM	Multi- agent approach	Deep LSTM	Proposed SSDCA- based Deep LSTM
Dataset 1	<i>RMSE</i>	0.2953	0.2229	0.1878	0.1165	0.1154	0.1114
	<i>MSE</i>	0.0872	0.0497	0.0353	0.0136	0.0133	0.0124
Dataset 2	<i>RMSE</i>	0.2971	0.2240	0.1892	0.1215	0.1143	0.1065
	<i>MSE</i>	0.0883	0.0502	0.0358	0.0148	0.0131	0.0113

8.5.5 Statistical Analysis

Table 8-9 shows the statistical analysis. In the statistical analysis, the proposed method's RMSE and MSE Mean are 0.1113, 0.0124, and the variance is 0.0002 using dataset-1. Conducting the same statistical analysis using dataset-2, the mean and variance of the proposed method were 0.1063, 0.0111 and 0.0002. The proposed method has a lower variance than the existing methods for RMSE and MSE.

Table 8-9 Statistical analysis

Dataset	Metrics	DBN	Gradient Boosting Classifier	Information Gain-SVM	Multi- Agent Approach	Deep LSTM	Proposed SSDCA-based Deep LSTM
Dataset- 1	<i>RMSE</i>	0.2953	0.2229	0.1878	0.1165	0.1154	0.1115
	<i>Mean</i>	0.2947	0.2225	0.1872	0.1162	0.1150	0.1113
	<i>Variance</i>	0.0006	0.0004	0.0006	0.0003	0.0004	0.0002
	<i>MSE</i>	0.0872	0.0497	0.0353	0.0136	0.0133	0.0124
	<i>Mean</i>	0.0868	0.0490	0.0350	0.0131	0.0130	0.0122
	<i>Variance</i>	0.0004	0.0007	0.0003	0.0005	0.0003	0.0002
Dataset- 2	<i>RMSE</i>	0.2971	0.2240	0.1892	0.1215	0.1143	0.1065
	<i>Mean</i>	0.2965	0.2236	0.1885	0.1210	0.1139	0.1063
	<i>Variance</i>	0.0006	0.0004	0.0007	0.0005	0.0004	0.0002
	<i>MSE</i>	0.0883	0.0502	0.0358	0.0148	0.0131	0.0113
	<i>Mean</i>	0.0878	0.0498	0.0353	0.0144	0.0126	0.0111
	<i>Variance</i>	0.0005	0.0004	0.0005	0.0004	0.0005	0.0002

8.5.6 Computational Time Analysis

We perform our experiment on a Personal Computer (PC) with an Intel(R) Core(TM) i7-9700 CPU with a processor speed of 3.00GHz and 32GHz RAM. We used libraries such as TensorFlow Core-2.4.1, TensorFlow GPU-2.4.1, NumPy-1.19.1, pandas-0.25.3, sci-kit learn-0.23.2, Scipy-1.5.2, PySimpleGUI-4.29.0 and Matplotlib-3.3.1. Table 8-10 shows the computational wall time. The

computational time analysis shows that our proposed SSDCA-based Deep LSTM needs 300sec and outperforms all other methods. Table 8-4 shows detailed computational time requirements.

Table 8-10 Computational time

Methods	Computational time (sec)
DBN	660.00
Gradient boosting classifier	690.00
Information gain-SVM	480.00
Multi-agent approach	630.00
Deep LSTM	420.00
Proposed SSDCA-based Deep LSTM	300.00

8.6 Conclusions and Future Work

Many factors can cause flight delays, ranging from failure in processes to late departure or aircraft arrival. The reasons for flight delays generate an enormous, complex amount of data used by Machine Learning methods to make crucial decisions because of the importance of flights arriving or departing on time for the airport, airlines and passengers. Developing flight delay prediction models with high accuracy is necessary. This paper proposes a novel optimised forecasting model with Deep LSTM for flight delay prediction. We utilise the developed SSDCA to train the Deep LSTM and the Deep RNN for fusing the features. The newly proposed SSDCA works by combining SSD and CAViaR algorithms. We employ the novel SSDCA to identify the best weights for effective flight delay prediction using the US Government Bureau of Transportation Statistics dataset.

Initially, we perform the pre-processing from the input data and then the data transformation based on the Yeo-Johnson transformation. Afterwards, we perform the feature fusion using the Deep RNN to extract the useful features from the original datasets containing complex and non-linear structures with spatial and temporal correlations. We introduced a two-step training procedure to help integrate the model's fused features and prediction layer. Finally, we apply the Deep LSTM for flight delay prediction. Here, we use an advanced optimisation

method named SSDCA to train the Deep LSTM and Deep RNN. We evaluate our proposed SSDCA-based Deep LSTM and compare it with four benchmark methods: DBN, gradient boosting classifier, information gain-SVM, multi-agent approach, and Deep LSTM, including other developed meta-heuristic approaches PSO+Deep LSTM, ACO+Deep LSTM, SSD+Deep LSTM, HBO+Deep LSTM and EWA+Deep LSTM. Our approach had a minimal RMSE and MSE of 0.1115 and 0.0124 on dataset-1, with 0.1065 and 0.0113 on dataset-2. The novel SSDCA-enabled approach performance has shown superior accuracy with a higher convergence rate than the other five meta-heuristic approaches regarding model accuracy and convergence analysis.

The results we get from our approach are promising. It illustrates how we can improve predicting flight delays using Deep Learning techniques with optimisation algorithms to inform departure and arrival policies and better management of airport facilities. At the same time, the stakeholders can attain efficiency and improve passenger satisfaction. The study can be extended further. One direction could be to consider more attributes. While there is no perfect delay prediction, in the future, we plan to investigate further flights with significant delays, even though they are rare events. At the same time, we keep the error range on a regular flight at a minimally acceptable level so that departing and arriving policies are developed with less complexity. Secondly, to investigate the performance, we will test our proposed technique on other data sets or sampling data from different sectors, such as maritime or rail. Finally, we will explore advanced optimisation approaches to improve the proposed method's performance towards a data-driven departure and arrival planning strategy, supporting airports, airlines and passengers to plan travel.

8.7 References

- [1] B. Yu, Z. Guo, S. Asian, H. Wang, and G. Chen, "Flight delay prediction for commercial air transport: A Deep Learning approach," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 125, no. March, pp. 203–221, 2019, doi: 10.1016/j.tre.2019.03.013.
- [2] A. Sternberg, J. Soares, D. Carvalho, and E. Ogasawara, "A Review on Flight Delay Prediction," pp. 1–21, 2017.
- [3] Y. Guleria, Q. Cai, S. Alam, and L. Li, "A Multi-Agent Approach for Reactionary Delay Prediction of Flights," *IEEE Access*, vol. 7, no. Cdm, pp. 181565–181579, 2019, doi: 10.1109/ACCESS.2019.2957874.
- [4] B. Alharbi and M. Prince, "A hybrid Artificial Intelligence approach to predict flight delay," *Int. J. Eng. Res. Technol.*, vol. 13, no. 4, pp. 814–822, 2020.
- [5] Z. Chen, P. Wanke, J. J. M. Antunes, and N. Zhang, "Chinese airline efficiency under CO2 emissions and flight delays: A stochastic network DEA model," *Energy Econ.*, vol. 68, pp. 89–108, 2017, doi: 10.1016/j.eneco.2017.09.015.
- [6] Ö. Şafak, A. Atamtürk, and M. Selim Aktürk, "Accommodating new flights into an existing airline flight schedule," *Transp. Res. Part C Emerg. Technol.*, vol. 104, no. April, pp. 265–286, 2019, doi: 10.1016/j.trc.2019.05.010.
- [7] C. Jiang and S. Zheng, "Airline baggage fees and airport congestion," *Transp. Res. Part C Emerg. Technol.*, vol. 117, no. July, p. 102686, 2020, doi: 10.1016/j.trc.2020.102686.
- [8] A. Dixit and S. K. Jakhar, "Airport capacity management: A review and bibliometric analysis," *J. Air Transp. Manag.*, vol. 91, no. February 2020, p. 102010, 2021, doi: 10.1016/j.jairtraman.2020.102010.
- [9] F. Chiti, R. Fantacci, and A. Rizzo, "An integrated software platform for airport queues prediction with application to resources management," *J. Air Transp. Manag.*, vol. 67, no. November 2017, pp. 11–18, 2018, doi: 10.1016/j.jairtraman.2017.11.003.
- [10] J. Chen and M. Li, "Chained predictions of flight delay using Machine Learning," *AIAA SciTech 2019 Forum*, no. January, pp. 1–25, 2019, doi: 10.2514/6.2019-1661.
- [11] M. Ball et al., "Total delay impact study: a comprehensive assessment of the costs and impacts of flight delay in the United States," *Nextor*, pp. 5–14, 25–30, 2010.
- [12] J. Chen and D. Sun, "Stochastic ground-delay-program planning in a metroplex," *J. Guid. Control. Dyn.*, vol. 41, no. 1, pp. 231–239, 2018, doi: 10.2514/1.G002964.
- [13] J. Ferguson, A. Q. Kara, K. Hoffman, and L. Sherry, "Estimating domestic US airline cost of delay based on European model," *Transp. Res. Part C Emerg. Technol.*, vol. 33, pp. 311–323, 2013, doi: 10.1016/j.trc.2011.10.003.
- [14] N. Pyrgiotis, K. M. Malone, and A. Odoni, "Modelling delay propagation within an airport network," *Transp. Res. Part C Emerg. Technol.*, vol. 27, pp. 60–75, 2013, doi: 10.1016/j.trc.2011.05.017.
- [15] Y. Ai, W. Pan, C. Yang, D. Wu, and J. Tang, "A Deep Learning approach to predict the spatial and temporal distribution of flight delay in network,"

- Journal of Intelligent and Fuzzy Systems*, vol. 37, no. 5. pp. 6029–6037, 2019, doi: 10.3233/JIFS-179185.
- [16] S. M. Al-Tabbakh, H. M. Mohamed, and Z. H. El, "Machine Learning Techniques for Analysis of Egyptian Flight Delay," *Int. J. Data Min. Knowl. Manag. Process*, vol. 8, no. 3, pp. 01–14, 2018, doi: 10.5121/ijdkp.2018.8301.
 - [17] J. Qu, T. Zhao, M. Ye, J. Li, and C. Liu, "Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data," *Neural Process. Lett.*, vol. 52, no. 2, pp. 1461–1484, 2020, doi: 10.1007/s11063-020-10318-4.
 - [18] F. Piccialli, N. Bessis, and J. J. Jung, "Guest Editorial: Data Science Challenges in Industry 4.0," *IEEE Trans. Ind. Informatics*, vol. 16, no. 9, pp. 5924–5928, 2020, doi: 10.1109/TII.2020.2984061.
 - [19] B. Thiagarajan, L. Srinivasan, A. V. Sharma, D. Sreekanthan, and V. Vijayaraghavan, "A Machine Learning approach for prediction of on-time performance of flights," *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, vol. 2017-Sept, 2017, doi: 10.1109/DASC.2017.8102138.
 - [20] G. C. Buttazzo, *Hard Real-Time Computing Systems*, vol. 24, no. 01. 2011.
 - [21] M. A. A. Makhloof, M. E. Waheed, and U. A. E. R. Badawi, "Real-time aircraft turnaround operations manager," *Prod. Plan. Control*, vol. 25, no. 1, pp. 2–25, 2014, doi: 10.1080/09537287.2012.655800.
 - [22] X. Zhu and L. Li, "Improved Flight Time Predictions for Fuel Loading Decisions of Scheduled Flights with a Deep Learning Approach," no. 11215119, pp. 1–12, 2020.
 - [23] S. Yao, *Flight Arrival Delay Prediction Using Gradient Boosting Classifier*, vol. 813, no. January. Springer Singapore, 2019.
 - [24] and V. T. Ball M. O., Chen C. Y., Hoffman R., *Collaborative decision making in air traffic management: Current and future research directions. In New Concepts and Methods in Air Traffic Management*. 2001.
 - [25] M. O. Ball, R. Hoffman, and A. Mukherjee, "Ground delay program planning under uncertainty based on the ration-by-distance principle," *Transp. Sci.*, vol. 44, no. 1, pp. 1–14, 2010, doi: 10.1287/trsc.1090.0289.
 - [26] A. K. Agogino and K. Tumer, "A multi-agent approach to managing air traffic flow," *Auton. Agent. Multi. Agent. Syst.*, vol. 24, no. 1, pp. 1–25, 2012, doi: 10.1007/s10458-010-9142-5.
 - [27] Y. Tu, M. O. Ball, and W. S. Jank, "Estimating flight departure delay distributions - A statistical approach with long-term trend and short-term pattern," *J. Am. Stat. Assoc.*, vol. 103, no. 481, pp. 112–125, 2008, doi: 10.1198/016214507000000257.
 - [28] E. R. Mueller and G. B. Chatterji, "Analysis of aircraft arrival and departure delay characteristics," *AIAA's Aircr. Technol. Integr. Oper. 2002 Tech. Forum*, no. October, pp. 1–14, 2002, doi: 10.2514/6.2002-5866.
 - [29] A. Klein, C. Craun, and R. S. Lee, "Airport delay prediction using Weather-Impacted Traffic Index (WITI) model," *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, pp. 1–13, 2010, doi: 10.1109/DASC.2010.5655493.
 - [30] A. J. Holden *et al.*, "Reducing the Dimensionality of," vol. 313, no. July, pp. 504–507, 2006.
 - [31] Y. Lecun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no.

- 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [32] B. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks. Imagenet classification with deep convolutional neural networks.,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2012.
 - [33] I. Ahmed, M. Ahmad, G. Jeon, and F. Piccialli, “A Framework for Pandemic Prediction Using Data Analytics,” *Data Res.*, vol. 25, p. 100190, 2021, doi: 10.1016/j.bdr.2021.100190.
 - [34] M. Hesam, N. Sultana, H. Millward, and L. Liu, “Ensemble learning activity scheduler for activity based travel demand models,” *Transp. Res. Part C*, vol. 123, no. February, p. 102972, 2021, doi: 10.1016/j.trc.2021.102972.
 - [35] A. Farrokhi, R. Farahbakhsh, J. Rezazadeh, and R. Minerva, “Application of Internet of Things and Artificial Intelligence for smart fitness: A survey,” *Comput. Networks*, p. 107859, 2021, doi: 10.1016/j.comnet.2021.107859.
 - [36] D. Truong, “Using causal Machine Learning for predicting the risk of flight delays in air transportation,” *J. Air Transp. Manag.*, vol. 91, no. May 2020, p. 101993, 2021, doi: 10.1016/j.jairtraman.2020.101993.
 - [37] N. Sharma, R. Sharma, and N. Jindal, “Machine Learning and Deep Learning Applications-A Vision Neha,” *Glob. Transitions Proc.*, pp. 0–8, 2021, doi: 10.1016/j.gltp.2021.01.004.
 - [38] J. C. Bansal, H. Sharma, S. S. Jadon, and M. Clerc, “Spider Monkey Optimisation algorithm for numerical optimisation,” *Memetic Comput.*, vol. 6, no. 1, pp. 31–47, 2014, doi: 10.1007/s12293-013-0128-0.
 - [39] S. Chander, V. Padmanabha, and J. Mani, “Jaya Spider Monkey Optimisation-driven Deep Convolutional LSTM for the prediction of COVID’19,” *Bio-Algorithms and Med-Systems*, vol. 16, no. 4. 2020, doi: 10.1515/bams-2020-0030.
 - [40] M. Guvercin, N. Ferhatosmanoglu, and B. Gedik, “Forecasting Flight Delays Using Clustered Models Based on Airport Networks,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–11, 2020, doi: 10.1109/tits.2020.2990960.
 - [41] M. Lambelho, M. Mitici, S. Pickup, and A. Marsden, “Assessing strategic flight schedules at an airport using Machine Learning-based flight delay and cancellation predictions,” *J. Air Transp. Manag.*, vol. 82, no. September 2019, p. 101737, 2020, doi: 10.1016/j.jairtraman.2019.101737.
 - [42] S. Pathomsiri, A. Haghani, M. Dresner, and R. J. Windle, “Impact of undesirable outputs on the productivity of US airports,” *Transp. Res. Part E Logist. Transp. Rev.*, vol. 44, no. 2, pp. 235–259, 2008, doi: 10.1016/j.tre.2007.07.002.
 - [43] M. Ankan, V. Deshpande, and M. G. Sohoni, “Building reliable air-travel infrastructure using empirical data and stochastic models of airline networks,” *52nd AGIFORS Annu. Proc. 2012 - Symp. Study Gr. Meet.*, vol. 2, no. June 2020, pp. 1107–1142, 2012.
 - [44] J. K. Mandal, *Emerging Technologies in Data Mining*, vol. 2. 2002.
 - [45] J. J. Rebollo and H. Balakrishnan, “Characterisation and prediction of air traffic delays,” *Transp. Res. Part C Emerg. Technol.*, vol. 44, pp. 231–241, 2014, doi: 10.1016/j.trc.2014.04.007.
 - [46] S. Choi, Y. J. Kim, S. Briceno, and D. Mavris, “Prediction of weather-induced airline delays based on Machine Learning algorithms,” in

- AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, vol. 2016-Decem, doi: 10.1109/DASC.2016.7777956.
- [47] J. V. Pérez-Rodríguez, J. M. Pérez-Sánchez, and E. Gómez-Déniz, "Modelling the asymmetric probabilistic delay of aircraft arrival," *J. Air Transp. Manag.*, vol. 62, pp. 90–98, 2017, doi: 10.1016/j.jairtraman.2017.03.001.
 - [48] Y. J. Kim, S. Choi, S. Briceno, and D. Mavris, "A Deep Learning approach to flight delay prediction," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, vol. 2016-Decem, pp. 1–6, doi: 10.1109/DASC.2016.7778092.
 - [49] M. F. Yazdi, S. R. Kamel, S. J. M. Chabok, and M. Kheirabadi, "Flight delay prediction based on Deep Learning and Levenberg-Marquart algorithm," *J. Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00380-z.
 - [50] E. Esmaeilzadeh and S. Mokhtarimousavi, "Machine Learning Approach for Flight Departure Delay Prediction and Analysis," *Transp. Res. Rec.*, vol. 2674, no. 8, pp. 145–159, 2020, doi: 10.1177/0361198120930014.
 - [51] D. A. Pamplona, L. Weigang, A. G. De Barros, E. H. Shiguemori, and C. J. P. Alves, "Supervised Neural Network with multilevel input layers for predicting of air traffic delays," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2018-July, 2018, doi: 10.1109/IJCNN.2018.8489511.
 - [52] H. M. Gomes *et al.*, "Adaptive random forests for evolving data stream classification," *Mach. Learn.*, vol. 106, no. 9–10, pp. 1469–1495, 2017, doi: 10.1007/s10994-017-5642-8.
 - [53] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transp. Res. Part C Emerg. Technol.*, vol. 54, pp. 187–197, 2015.
 - [54] Y. Liu, G. Jia, X. Tao, X. Xu, and W. Dou, "A stop planning method over big traffic data for airport shuttle bus," *Proc. - 4th IEEE Int. Conf. Data Cloud Comput. BDCloud 2014 with 7th IEEE Int. Conf. Soc. Comput. Networking, Soc. 2014 4th Int. Conf. Sustain. Comput. C*, pp. 63–70, 2014, doi: 10.1109/BDCloud.2014.21.
 - [55] N. Xu, G. Donohue, K. B. Laskey, and C. H. Chen, "Estimation of delay propagation in the national aviation system using Bayesian networks," *Proc. 6th USA/Europe Air Traffic Manag. Res. Dev. Semin. ATM*, pp. 353–363, 2005.
 - [56] G. Tsiotas, "On the use of non-linear transformations in Stochastic Volatility models," *Stat. Methods Appl.*, vol. 18, no. 4, pp. 555–583, 2009, doi: 10.1007/s10260-008-0113-9.
 - [57] M. Inoue, S. Inoue, and T. Nishida, "Deep recurrent neural network for mobile human activity recognition with high throughput," *Artif. Life Robot.*, vol. 23, no. 2, pp. 173–185, 2018, doi: 10.1007/s10015-017-0422-x.
 - [58] A. Tharwat and T. Gabel, "Parameters optimisation of support vector machines for imbalanced data using social ski driver algorithm," *Neural Comput. Appl.*, vol. 32, no. 11, pp. 6925–6938, 2020, doi: 10.1007/s00521-019-04159-z.
 - [59] R. F. Engle and S. Manganelli, "CAViaR: Conditional autoregressive value at risk by regression quantiles," *J. Bus. Econ. Stat.*, vol. 22, no. 4, pp. 367–381, 2004, doi: 10.1198/073500104000000370.

- [60] S. Kim, S. Hong, M. Joh, and S. K. Song, "DeepRain: ConvLSTM network for precipitation prediction using multichannel radar data," *arXiv*, pp. 3–6, 2017.
- [61] "January Flight Delay Prediction dataset taken from, 'https://www.kaggle.com/divyansh22/flight-delay-prediction', accessed on July 2020.
- [62] "February Flight Delay Prediction dataset taken from, 'https://www.kaggle.com/divyansh22/february-flight-delay-prediction#Feb_2020_ontime.csv', accessed on July 2020."
- [63] M. C. Thrun and A. Ultsch, "Swarm intelligence for self-organised clustering," *Artif. Intell.*, vol. 290, p. 103237, 2021, doi: 10.1016/j.artint.2020.103237.
- [64] J. Zhou *et al.*, "Optimisation of support vector machine through the use of metaheuristic algorithms in forecasting TBM advance rate," *Eng. Appl. Artif. Intell.*, vol. 97, no. October 2020, p. 104015, 2021, doi: 10.1016/j.engappai.2020.104015.
- [65] T. S. Lim, W. Y. Loh, and Y. S. Shih, "Comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms," *Mach. Learn.*, vol. 40, no. 3, pp. 203–228, 2000, doi: 10.1023/A:1007608224229.
- [66] D. B. Bisandu, M. S. Homaid, I. Moulitsas, and S. Filippone, "A deep feedforward neural network and shallow architectures effectiveness comparison: flight delays classification perspective," *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK*, 2021, (Accepted).
- [67] The MathWorks, I. (2018b). Symbolic Math Toolbox. Natick, Massachusetts, United States. Retrieved from <https://www.mathworks.com/help/symbolic/>.
- [68] Z. Guo, B. Yu, M. Hao, W. Wang, Y. Jiang, and F. Zong, "A novel hybrid method for flight departure delay prediction using Random Forest Regression and Maximal Information Coefficient," *Aerospace Science and Technology*, Vol. 116, 106822, 2021.
- [69] G. Vonitsanos, T. Panagiotakopoulos, Kanavos, A., and A. Tsakalidis, "Forecasting Air Flight Delays and Enabling Smart Airport Services in Apache Spark," In *IFIP International Conference on Artificial Intelligence Applications and Innovations* (pp. 407-417). Springer Cham, 2021.
- [70] R. Dalmau, F. Ballerini, H. Naessens, S. Belkoura, and S. Wangnick, "An explainable Machine Learning approach to improve take-off time predictions," *Journal of Air Transport Management*, no. 95, 102090, 2021.
- [71] W. Shao, P. Arian, Z. Sichen, K. Piotr, and D. S. Flora, "Predicting Flight Delay with Spatio-Temporal Trajectory Convolutional Network and Airport Situational Awareness Map." *arXiv preprint arXiv:2105.08969*, 2021.
- [72] X. Zhu, and L. Li, "Flight time prediction for fuel loading decisions with a Deep Learning approach," *Transportation Research Part C: Emerging Technologies*, no. 128, 103179, 2021.
- [73] S. Somani, P. Priyanshu, S. Meghna, and M. Safa. "An Approach of Applying Machine Learning Model in Flight Delay Prediction-A Comparative Analysis," *Revista Geintic-Gestao Inovacao E Techlogias*,

- vol. 11, no. 3, pp. 1233-1244, 2021.
- [74] M. Y. Aghdam, S. R. K. Tabbakh, S. J. M. Chabok, and K. Maryam "Optimisation of air traffic management efficiency based on Deep Learning enriched by the long short-term memory (LSTM) and extreme learning machine (ELM)," *Journal of Data* 8, no. 1 pp. 1-26, 2021.
 - [75] M. Zoutendijk, and M. Mihaela "Probabilistic Flight Delay Predictions Using Machine Learning and Applications to the Flight-to-Gate Assignment Problem," *Aerospace* vol. 8, no. 6, pp.152, 2021.
 - [76] H. Chen, S. Tu, and H. Xu, "The Application of Improved Grasshopper Optimisation Algorithm to Flight Delay Prediction–Based on Spark," In *Conference on Complex, Intelligent, and Software Intensive Systems* (pp. 80-89). Springer, Cham, 2021.
 - [77] L. Giusti, L. Carvalho, A. T. Gomes, R. Coutinho, J. Soares, and E. Ogasawara, "Analysing Flight Delay Prediction Under Concept Drift,". *arXiv preprint arXiv:2104.01720*, 2021.
 - [78] W. Chen, Z. Li, C. Liu, and Y. Ai, "A Deep Learning Model with Conv-LSTM Networks for Subway Passenger Congestion Delay Prediction," *Journal of Advanced Transportation*, 2021.
 - [79] W. A. Khan, H. L. Ma, S. H. Chung, and X. Wen, "Hierarchical integrated Machine Learning model for predicting flight departure delays and duration in series," *Transportation Research Part C: Emerging Technologies*, 129, 103225, 2021.
 - [80] H. Alla, L. Moumoun, and Y. Balouki, "A Multilayer Perceptron Neural Network with Selective-Data Training for Flight Arrival Delay Prediction," *Scientific Programming*, 2021.

Chapter 9: Prediction of Flight Delay using Deep Operator Network with Gradient-Mayfly Optimisation Algorithm

Accurate flight delay prediction is fundamental to establishing an efficient airline business. It is considered one of the most critical intelligent aviation systems components. Recently, flight delay has been a significant cause that deprives airlines of good performance. Hence, airlines must accurately forecast flight delays and comprehend their sources to have excellent passenger experiences, increase income and minimise unwanted revenue loss. In this paper, we developed a novel approach that is an optimisation-driven Deep Learning model for predicting flight delays by extending a state-of-the-art method, DeepONet. We utilise the Box-Cox transformation for data conversion with a minimal error rate.

Also, we employed a deep residual network for the feature fusion before training our model. Furthermore, this research uses flight on-time data for flight delay prediction. To validate our proposed model, we conducted a numerical study using the US Bureau of Transportation of Statistics. Also, we predict the flight delay by selecting the optimum weights using the novel DeepONet with the Gradient Mayfly Optimisation Algorithm (GMOA). Our experiment results show that the proposed GMOA-based DeepONet outperformed the existing methods with a Root Mean Square Error of 0.0765, Mean Square Error of 0.0058, Mean Absolute Error of 0.0049 and Mean Absolute Percent Error of 0.0043, respectively. These results also show the importance of optimisation algorithms in deciding the optimal weight to improve the model performance. The efficacy of our proposed approach in predicting flight delays with minimal errors well define from all the evaluation metrics. Also, utilising the prediction outcome of our robust model to release information about the delayed flight in advance from the aviation decision systems can effectively alleviate the passengers' nervousness.

9.1 Introduction

The performance of air transportation operation systems has been affected by flight delays leading to substantial economic losses and adversely affecting the planning of passengers, airlines and airports [1–3]. Flight delay is recognised when discrepancies exist between an aircraft's schedule or arrival and actual departure and arrival time [4–6]. Flights delay contributes to the advantages and disadvantages of airlines, airports and passengers and is unavoidable [7–9]. In 2013 31.1% of flights in the

United States were reported to be delayed by 15 minutes. Also, 36% of European flights were delayed by 5 minutes, and 16.3% of Brazil flights were delayed by 30 minutes [10]. These reports show the importance of the flight delay topic and its effects [5–11]. The estimated sum of \$7.4 million is the annual cost in China caused by the flight delay [10]. Flight delays in 2007 have a cost implication estimated at \$ 32.9 in the economy of the United States of America (USA) [12–13]. The practical impact of flight delays can affect the passenger experience and performance of the airport. Passengers react with negative emotional dominance, especially service failures [10]. Some causes of flight delays are security, equipment problems and flight crew delays [10, 12]. The delay affected the trade as improved transport relies on customers' trust to minimise or increase ticket sales, so that time flight led to customer self-reliance [2–4]. Thus, flight prediction can lead to an expert decision and functioning for airports and agencies and gratify the client [16, 17, 26, 27, 18–25, 28]. Recently, flight delay has become a severe issue for several travellers and the world's civil aviation industries. To solve the problems of flight delays, the Civil Aviation Administration has devised several programs for reducing the average flight delay time to improve the effectiveness of flight operations. The significant aspect refers to the airport capacity and airspace being inadequate. Several reasons, like airport scheduling, company plans, and luggage, caused delays [6–8].

Despite comprehending the spatial dependence amongst airports on a flight delay, the previous techniques focused on evaluating delays' qualitative impact and are unsuitable for forecasts [9–12]. Even though some of the analyses were produced to predict flight delay, they did not meet the needs of the realistic application. There exist two major issues in prior techniques. The classical techniques consider small sample sizes, so enhancing the prediction accuracy is complex. It is essential to minimise airport dimensions as it can cause data loss [13–15]. Several sources activate the flight delay, impacting the airlines, airports and on-the-way paths. Some airport techniques combine destination and airlines to analyse the data to forecast flight delays. The airport poses the top precedence of examination on the efficiency using delays produced by carriers. The link between delay and congestion is inspected in an ensemble of routes and airports. Other airlines and airports should be adapted for computing capacity issues and deciding the airlines. Several ways exist to evaluate the complete scope, and it is essential to adapt the air transport models, particularly if the delays in the root are troubled [16–18].

Various Machine Learning models are devised in prior works to predict flight delays. Most studies predicted flight delays using binary classifiers, multi-class and regression classifiers. The binary classifier reveals if the flight is delayed or not delayed. The multi-class classifiers produce several delay classes. The regression classifier estimates the value of delay [4, 19, 20]. Most of the research is proposed to address the issue of predicting flight delays with classical Machine Learning methods. Classical Machine Learning techniques, like decision trees, Bayesian networks and stochastic forests, are introduced for predicting flight delay [4, 13, 21, 22]. Deep models are developed to capture spatial and temporal dependencies to predict flight delays. Predicting precise traffic has acquired focus in managing air transportation because of several advantages. Here, a graph convolutional neural network (GCN) poses the ability to mine complicated non-linear relations in graphs, bringing prospects in managing complex traffic prediction issues using the deliberation of graph-structured data. The authors have devised intellectual techniques based on GCN models to offer quantified transportation diagnostics [23–26]. In [49], a recurrent neural network is designed for modelling the sequential data to predict traffic. In [50], a Meta recurrent neural network is integrated with an encoder-decoder model for predicting urban traffic. In [51], improved GCN is devised by adapting the attention technique for collecting neighbours' features and developing a deep model for predicting the traffic.

Our research presents a novel approach optimisation-aware deep model to forecast flight delays by extending a state-of-the-art DeepONet method. Considering the weights and biases of the Deep Residual Network (DRN), we perform the feature fusion by selecting the value of the optimal coefficient. The weight and bias update of DRN is achieved with Political Social Ski Driver Conditional Autoregressive (PSSDCA), which is generated by integrating social ski driver (SSD), CAViaR and Political Optimisation (PO). In addition, the flight delay prediction is made with the proposed Gradient Mayfly Optimisation Algorithm (GMOA) based DeepONet. The weight update issue of DeepONet is solved with the proposed GMOA, wherein the weight update in training exploited GMOA, which is the combination of MA and gradient descent algorithm.

The major contributions of the paper include:

- i. **Proposed PSSDCA-based DRN for fusing features:** The fusion of features is performed with a designed PSSDCA-based DRN. The features are employed on DRN to perform feature fusion with weights and biases based on hidden

layers. The DRN training is performed using the proposed PSSDCA and obtained by combining SSD, CAViaR and PO by tuning optimum weights.

- ii. **Proposed GMOA-based DeepONet to predict flight delay:** The prediction of flight delay is performed using the proposed GMOA-based DeepONet. The weight update of DeepONet is achieved using the proposed GMOA by training the DeepONet. The MA is integrated with a gradient descent algorithm for selecting the best weights by minimising the error.

The rest of the sections are as follows: Section 9.2 discusses the literature review on flight delay prediction models. Section 9.3 describes the proposed model for the flight delay. Section 9.4 contains the results and discussions, while Section 9.5 presents lessons learned and the conclusion.

9.2 Literature Review

Several studies about airport flight delay prediction and management exist [39–46]. The flight delay prediction consists of arrival or departure time, depending on the interest of the study. The aim of the prediction is generally to predict the flight arrival or departure time accurately. The results of the predictions are utilised for decision-making. Thus, the accuracy of the prediction affects the entire system. Several classical forecasting methodologies and prediction theories have been mentioned in the literature, including the Naïve Bayes model, Support Vector Machine, Logistic Regression, multivariate regression and time series prediction [28]. However, these methods and theories may be less suitable as flight delays is a multidimensional time series with features of high complexity that need more sophisticated fusion procedure [47–49]. Therefore, we introduce a novel approach based on the state-of-the-art approach called DeepONet [62] integrated with an optimisation algorithm to improve the training and prediction performance of the model by reducing the error to the barest minimum. In particular, the input is a function $f(x)$, including all the important factors that lead to delays. Predicting flight delays is easier based on the flight dynamics, neural network, and real-world data collected from the BTS. Additionally, the model performance is improved by the pre-processing procedure proposed.

The eight conventional flight delay prediction methods and their merits and demerits are enlisted in Güvercin M et al. [63]; the authors devised clustered airport modelling techniques to predict the delays in flight considering airport networks. The technique offered a precise prediction of flight delays. However, the method did not

choose imperative features to forecast delay. Lambelho Met al. [31] devised a generic analysis of strategic schedules at an airport with a flight delay to include imperative features. The method offered good efficiency, but this technique did not provide improved prediction accuracy. To enhance accuracy, Chen Zet al. [64] developed the Information Gain-Support Vector Machine method for discovering the Chinese airline efficiency under Carbon dioxide (CO₂) emissions and flight delays. The method minimised the issues of the classical Data Envelopment Analysis (DEA) model but did not offer more policy implications. Chen J and Li M [6] devised a Machine Learning technique for predicting flight delays to offer policy implications. The method offered a precise model for delay prediction but did not involve an effectual database to enhance the accuracy.

To handle another database, Alla et al. [5] utilised artificial neural networks (ANN) and Multilayer perceptron (MLP) for predicting flight delay. Here, the authors extract chronological flight data to enhance efficiency. The technique offered better accuracy but was unsuitable for real-time flight data. To deal with real-time data, Yi Jet al. [65] developed a standard Stacking classification technique for predicting flight delay. The method effectively dealt with the imbalanced datasets, and the Boruta algorithm was used to choose effectual features. However, the method did not include other Machine Learning methods to learn flight delays. Guo et al. [66] devised a hybrid model, Random Forest Regression and Maximal Information Coefficient (RFR-MIC), to include Machine Learning methods for predicting flight delay. The Regression Maximal Information Coefficient and Random Forest were combined to predict the delay. However, the technique did not use cloud or parallel computing to minimise the computation time. To minimise the computation time, Shao, Wet al. [67] devised a vision-based solution for predicting flight delay. A deep end-to-end model, TrajCNN, was utilised to capture spatial and temporal data using the situational awareness map. However, the method was not applicable for long-term prediction.

Bisandu et al. [17] developed a novel method for an accurate flight forecast using social ski driver and conditional autoregressive value at risk. Their proposed model outperforms existing models and meta-heuristic approaches in predicting flight delays. However, the authors recommend other approaches to be developed based on other optimisation algorithms compared with their proposed models' performance. In [14], Bisandu et al. design a model for predicting flight delay using a deep feedforward network. Their model's performance is compared to shallow architectures, such as

support vector machines and single-layer neural networks. Their model outperforms the shallow architectures regarding the accuracy and classification of flight delays. Kim et al. [37] proposed different architectural designs and implementation of LSTM and RNN in predicting flight delays using sequences of thresholds. Lin et al. [38] the authors performed airport flight delay prediction using convolutional LSTM (Conv-LSTM) using temporal and spatial characteristics in China civil aviation. Their research shows the importance of parameter tuning when dealing with huge datasets and deep-learning neural networks.

The paper proposes a novel method by extending the state-of-the-art DeepONet, flight time reliability and airport network disruption literature. The contribution listed in the introduction can be emphasised as follows: First, we indicated the importance of reliability to all the aviation industry stakeholders. Second, we demonstrate how single airline disruption can negatively affect passenger satisfaction and cause airlines huge losses and flight itinerary schedules. Third, our novel model shows superior results through our extensive experiments by reducing the delay time to the barest minimum. The evaluation results using the different error measures in our models reveal how important having a small error of prediction on all the metrics and the lessons the passenger and airline industry can learn from predicting, mitigating and navigating flight delays in the future.

9.2.1 Problem Definition

According to the Federal Aviation Administration (FAA), any flight that departs or arrives 15 minutes later than the scheduled time is considered late or delayed. Several factors cause this delay at the origin or destination airports. The FAA has categorised the delays as NAS, carrier, weather, security and late aircraft. Assume T is the scheduled time for departure or arrival of flight F . Flight F is considered delayed if it takes off or arrives at the Origin-Destination airport at a time $T > 15$ minutes. A delay is considered severe based on how later than 15 minutes the flight arrives or departs; a delay is small if it is as small as $T = 5$ minutes to $T = 20$ minutes or severe if it is up to $T = 65$ minutes. All these factors play a significant role in airport departure or arrival delays; a flight may encounter a mild or severe delay under any given conditions. Therefore, this research aims to develop a novel deep-learning approach using DeepONet to predict flight delay for a month.

9.3 Proposed GMOA-based DeepONet for Prediction of Flight Delay

The delay in flights hurts passengers and airlines. The forecasting of delays is essential in aviation. In addition, the design of precise prediction techniques for predicting flight delays becomes complex because of complications of air transportation models and the accumulation of flight data. The goal is to devise a technique for predicting flight delay considering GMOA-based DeepONet. Initially, the input data is subjected to data pre-processing to discard the data redundancies from the data. After that, the data transformation is done using Box-Cox transformation to make it suitable for improved processing. The feature fusion is done with DRN, wherein the training of DRN is done with PSSDCA. The PSSDCA is obtained by integrating SSD, CAViaR and PO. The flight delay prediction is performed with DeepONet, trained with GMOA. The proposed GMOA combines Stochastic Gradient Descent (SGD) and the mayfly algorithm (MA). Figure 9-1 shows the structure of the flight delay prediction model using the proposed GMOA-based DeepONet.

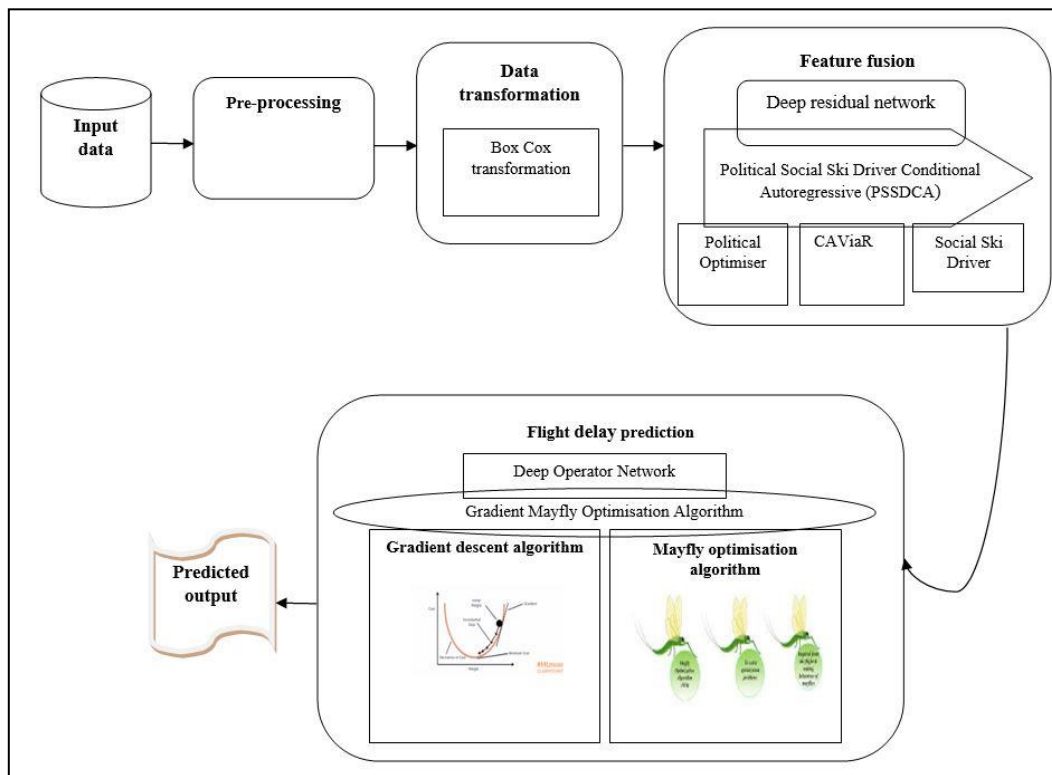


Figure 9-1 Structure of Flight Delay Prediction Model using Proposed Gmoa-Based Deeponet

Consider a database M in which the input data is expressed as Z considering various attributes, which is given by Equation (9-1):

$$M = \{Z_{f,g}\}; (1 \leq f \leq F); (1 \leq g \leq G) \quad (9-1)$$

where, $Z_{f,g}$ signifies data present in dataset M signifying g^{th} attribute of f^{th} data and F specify total data, and G signifies total attributes.

9.3.1 Pre-processing for Noise Removal

Here, the input data $Z_{f,g}$ is employed to undergo pre-processing. Pre-processing is a necessary step to inspect the data. The significance of pre-processing is to offer smooth processing of input data. The pre-processing is done to make the data suitable for enhanced processing. Thus, the output generated through pre-processed data is expressed as $(H_{f,g})_{J \times K}$ in which the matrix gives pre-processed data dimension $[J' \times K']$.

9.3.2 Data Transformation with Box cox Transformation

The pre-processed data $H_{f,g}$ Obtained from pre-processing is subjected to data transformation wherein Box-Cox transformation is adapted to perform a specific task. Data transformation is a procedure to transform data from one structure to another. Thus, the data transformation is adapted to organize data for improved processing. The transformation assists in enhancing knowledge quality. Box-Cox transformation [68] transforms non-normal data into some structure of data that undergoes normal distribution. The Box-Cox transformation is applied to pre-processed data $H_{f,g}$ and is expressed by Equation (9-2):

$$Q = \begin{cases} \frac{(H_{f,g})^\alpha - 1}{\alpha}; \alpha \neq 0 \\ \ln(H_{f,g}); \alpha = 0 \end{cases} \quad (9-2)$$

where Q is the outcome obtained from data transformation, and α is the transformation parameter. Thus, the output of data transformation is denoted as Q whose dimension is given by the matrix $[J \times K]$.

9.3.3 Fusion of Features with Developed PSSDCA-based DRN

After transforming data, the fusion of features is carried out with DRN [69], which is trained by developed PSSDCA using the transformed data Q . The PSSDCA is the incorporation of PO [70], SSD [71] and CAViaR [72]. After performing the transformation of data, the data dimension is $J \times K$, and thus, for minimising the features, the feature fusion step is essential to discover flight delay in an effectual manner. For instance, the G attributes have n columns; for example, n features are there and is given as $\{d_1, \dots, d_n\}$.

Sorting of Features with Correlation

Here, the correlation is calculated for the five features. For instance, the correlation is evaluated using d_1 and target values. Hence, correlation values are generated. After that, the columns are altered, considering high values of correlation. Thus, to minimise the features, the fusion of features is performed.

Feature Fusion

After the alteration of columns with correlation values, the fusion of features is done using the equation formulated as in Equation (9-3):

$$d_t^{fused} = \sum_{r=1}^L \frac{\eta}{h} d_t \quad (9-3)$$

Where $r = 1 + \frac{Q}{u}$, $h = 1$ to u , Q is total features, u signifies the count of features to be chosen, and is given by Equation (9-4):

$$u = \frac{Q}{L}; 1 \leq t \leq L \quad (9-4)$$

where t is the fused features index.

Discovery of η Considering DRN

Once the feature fusion is done, η is discovered using DRN. The features are adapted on DRN to perform feature fusion with weights and biases based on hidden layers.

a) Structure of DRN

Generally, the deep model offers improved accuracy and efficiency in generalizing data. In addition, the deep model can deduce features automatically and can be optimally tuned to attain enhanced efficiency. In addition, the deep model is robust and generates results quickly. The DRN [69] is a promising deep neural network model in Deep Learning and is adapted to various pattern recognition processes. DRN poses advantages like elevated training speed, the easy transmission of Gradient, etc. The DRN can be employed to attain elevated efficiency in both regression and classification procedures. The DRN comprises various layers like the convolutional (Conv) layer, pooling layer, activation function, batch normalization, and residual blocks. Figure 9-2 displays the structure of DRN.

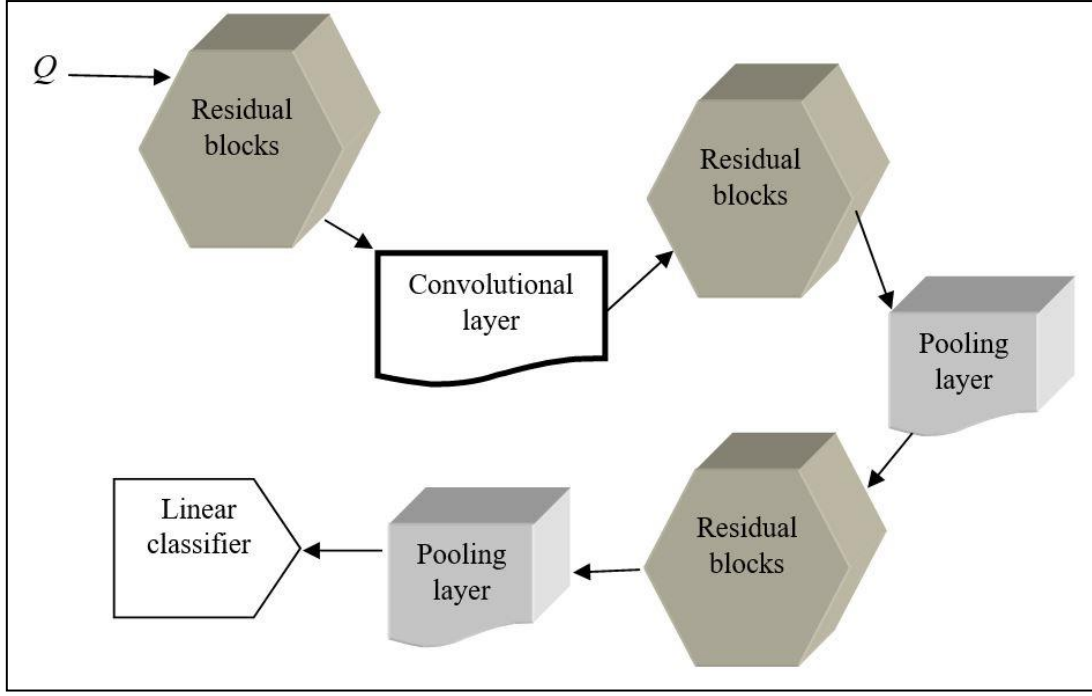


Figure 9-2 Architectural Model of DRN

(i) Conv layer

The Conv layer helps to extract effectual features automatically, considering inputted feature maps with a group of small amenable domains known as kernels. The 2D Conv layer (Conv2d) can reduce free attributes in the training process and improve the efficiency of the local receptive field and weight sharing. The Conv2d operates with the input using a sequence of filters termed kernels in a small domain with local connections. The cross-correlation of Conv2d is formulated as Equation (9-5) and Equation (9-6):

$$Conv2d(I_r) = \sum_{w=0}^{u-1} \sum_{z=0}^{v-1} Y_{w,z} \bullet I_{i_{(u+w),(v+z)}} \quad (9-5)$$

$$Conv1d(I_r) = \sum_{t=0}^{c_{in}-1} Y_z \ast I_u \quad (9-6)$$

Here, I_r refers to input, and I_i and I_u are input with the sequence of filters, u and v which are used for recording coordinates, Y is $u \times u$ kernel matrix and is known as a learnable attribute in the process of training, w and z express position index of 2-D kernel matrix, Y_z is kernel size of z^{th} input neuron and $\ast$ is a cross-correlation operator without zero padding.

(ii) Pooling layer

Here, the dimension evaluation is the same as that of Conv1d and Conv2d. The pooling layer is placed amongst two successive convolution layers, and it is commonly

adapted to minimise the spatial size of feature maps and overwhelms the overfitting issue effectively. For functioning with individual depth and a slice of the feature map, average pooling is chosen and is given by Equation (9-7) and Equation (9-8):

$$w_{out} = \frac{w_{in} - t_w}{B} + 1 \quad (9-7)$$

$$z_{out} = \frac{z_{in} - t_h}{B} + 1 \quad (9-8)$$

where, w_{in} and z_{in} is the width and height of the input 2D matrix, w_{out} and z_{out} express as output. In addition, the height and width of kernel size are expressed as t_w and t_h .

(iii) Activation function

The activation function is utilised for learning complicated non-linear features with the original dataset and is adapted to reduce the non-linearity of mined features. The Rectified Linear Unit (ReLU) is selected as a non-linear activation function, which is given below. Here, I_r signifies input feature maximises convergence rate and prevents vanishing gradient issue. It can be represented in Equation (9-9):

$$ReLU(I_r) = \begin{cases} 0, & I_r < 0 \\ I_r, & I_r \geq 0 \end{cases} \quad (9-9)$$

(iv) Batch normalization

The batch normalization (Batch Norm) is devised for minimising the internal covariate shift by handling the input layers with scaling. This maximises the rate of learning and also vanishes gradient issues. The input I_r comprises several counts of mini-batch samples and μ_c and σ_c refers to the mean and variance of mini-batch samples. The attributes v and ρ are employed as two learning attributes.

(v) Residual blocks

The residual block is formatted with a shortcut correlation between the input and output. Here, the inputs are linked to outputs if both acquire the same dimension. The dimension matching factor is given to matching input and output dimensions when the dimension is dissimilar, as shown by Equations (9-10) and (9-11), respectively.

$$A_1 = \zeta(I_r) + I_r \quad (9-10)$$

$$A_1 = \zeta(I_r) + \delta \cdot I_r \quad (9-11)$$

where, I_r and A_1 is input and output of residual block, ζ signifies mapping relationship functions amongst inputs and outputs, and δ refers to dimension matching factor. Thus, the output obtained from DRN is denoted as E .

b) Training data

The training data is essential for developing the model. After the discovery of η , the value of the training data is evaluated by correlating the means of data that belongs to a specific class. Table 9-1 exposes the discovery of training data.

Table 9-1 Discovery of training data

k_l $/d_Q$	d_1	d_2	d_3	d_Q	Target
k_1	$C(k_1, d_1)$	$C(k_1, d_2)$	$C(k_1, d_3)$	$C(k_1, d_Q)$	$\eta_{training}$
k_2	$C(k_2, d_1)$	$C(k_2, d_2)$	$C(k_2, d_3)$	$C(k_2, d_Q)$	$\eta_{training}$
k_3	$C(k_3, d_1)$	$C(k_3, d_2)$	$C(k_3, d_3)$	$C(k_3, d_Q)$	$\eta_{training}$
k_4	$C(k_4, d_1)$	$C(k_4, d_2)$	$C(k_4, d_3)$	$C(k_4, d_Q)$	$\eta_{training}$

The training of η is evaluated as in Equation (9-12):

$$\eta_{training}^{(r)} = \text{Correlation}(k_r, D_{k_r}) \quad (9-12)$$

where, D_{k_r} is the mean vector of data k_r that belongs to a class.

c) Training of DRN using PSSDCA

The training of DRN to find η is done using the proposed PSSDCA wherein the proposed PSSDCA is devised by incorporating PO in SSDCA. SSDCA acquires the benefits of both SSD and CaViaR by providing an enhanced balance between exploitation and exploration. The method addressed highly non-linear problems using complex constraints. It helped to address multi-objective optimisation issues. Meanwhile, PO [70] is motivated by the multiple phases of politics. The PO models the main stages of politics, like allocation of a constituency, switching of parties, election campaign, and election of inter-party and parliamentary affairs. It applies to real-world tools and can solve various engineering optimisation issues. It has a good speed of convergence and improved exploration ability in early iterations. Thus, incorporating PO in SSDCA improves the overall performance and attains a globally optimum solution. The steps of developing PSSDCA are:

Step 1) Initialisation:

The first step is solution initialisation, which is given by Equation (9-13):

$$C = \{C_1, C_2, \dots, C_K, \dots, C_\theta\} \quad (9-13)$$

Where ϑ refers to total solution and C_k signifies k^{th} solution.

Step 2) Finding Error:

The best solution is chosen using error and is termed a minimisation issue. The minimal error reveals the best solution and is given by Equation (9-14):

$$MSE = \frac{1}{F} [\sum_{f=1}^F \varepsilon - E] \quad (9-14)$$

where ε refers to actual output, E indicates estimated DRN output, and F is total data.

Step 3) Discovery of update equation of proposed PSSDCA:

To balance the exploration and exploitation phase, the SSDCA is adapted. Here, the updated equation of SSDCA is given as in Equation (9-15):

$$C_{b,c}^{a+1} = \gamma_0 + \gamma_1 C_{b,c}^{a-1} + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2}) [1 - e\text{Sin}(l_1) - \text{Sin}(l_1)] + e\text{Sin}(l_1) N_{b,c}^a + \text{Sin}(l_1) P_{b,c}^a \quad (9-15)$$

where, l_1 and l_2 are random numbers from 0 to 1. $N_{b,c}^a$ indicates optimal agents solution, and $P_{b,c}^a$ is the mean global solution for the whole population, $C_{b,c}^{a-1}$, $C_{b,c}^{a-2}$ represents solution at iteration $(a-1)$ and $(a-2)$, γ_0 and γ_1 are constants and $f(\cdot)$ represent fitness function.

The PO is utilised to obtain enhanced optimisation ability and solve engineering issues. The updated equation as per PO is given as Equations (9-16) to (9-20):

$$C_{b,c}^{a+1} = C_{b,c}^{a-1} + m(C_{b,c}^a - C_{b,c}^{a-1}) \quad (9-16)$$

$$C_{b,c}^{a+1} = C_{b,c}^{a-1} + mC_{b,c}^a - mC_{b,c}^{a-1} \quad (9-17)$$

$$C_{b,c}^{a+1} = C_{b,c}^{a-1}(1 - m) + mC_{b,c}^a \quad (9-18)$$

$$C_{b,c}^{a+1} - mC_{b,c}^a = C_{b,c}^{a-1}(1 - m) \quad (9-19)$$

$$C_{b,c}^{a-1} = \frac{C_{b,c}^{a+1} - mC_{b,c}^a}{(1-m)} \quad (9-20)$$

Substitute Equation (9-20) in Equation (9-15), and simplify yields Equations (9-21) to (9-25):

$$C_{b,c}^{a+1} = \gamma_0 + \gamma_1 \left[\frac{C_{b,c}^{a+1} - mC_{b,c}^a}{(1-m)} \right] + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2}) [1 - e\text{Sin}(l_1) - \text{Sin}(l_1)] + e\text{Sin}(l_1) N_{b,c}^a + \text{Sin}(l_1) P_{b,c}^a \quad (9-21)$$

$$C_{b,c}^{a+1} = \gamma_0 + \frac{\gamma_1 C_{b,c}^{a+1}}{(1-m)} - \frac{\gamma_1 m C_{b,c}^a}{(1-m)} + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2})[1 - e \sin(l_1) - \sin(l_1)] + e \sin(l_1) N_{b,c}^a + \sin(l_1) P_{b,c}^a \quad (9-22)$$

$$C_{b,c}^{a+1} - \frac{\gamma_1 C_{b,c}^{a+1}}{(1-m)} = \gamma_0 - \frac{\gamma_1 m C_{b,c}^a}{(1-m)} + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2})[1 - e \sin(l_1) - \sin(l_1)] + e \sin(l_1) N_{b,c}^a + \sin(l_1) P_{b,c}^a \quad (9-23)$$

$$C_{b,c}^{a+1} \left[1 - \frac{\gamma_1}{(1-m)}\right] = \gamma_0 - \frac{\gamma_1 m C_{b,c}^a}{(1-m)} + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2})[1 - e \sin(l_1) - \sin(l_1)] + e \sin(l_1) N_{b,c}^a + \sin(l_1) P_{b,c}^a \quad (9-24)$$

$$C_{b,c}^{a+1} \left[\frac{1-m-\gamma_1}{(1-m)}\right] = \gamma_0 - \frac{\gamma_1 m C_{b,c}^a}{(1-m)} + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2})[1 - e \sin(l_1) - \sin(l_1)] + e \sin(l_1) N_{b,c}^a + \sin(l_1) P_{b,c}^a \quad (9-25)$$

The final update equation of the developed PSSDCA is given as in Equation (9-26):

$$C_{b,c}^{a+1} = \frac{(1-m)}{1-m-\gamma_1} \left[\gamma_0 - \frac{\gamma_1 m C_{b,c}^a}{(1-m)} + \gamma_2 C_{b,c}^{a-2} + \gamma_1 f(C_{b,c}^{a-1}) + \gamma_2 f(C_{b,c}^{a-2})[1 - e \sin(l_1) - \sin(l_1)] + e \sin(l_1) N_{b,c}^a + \sin(l_1) P_{b,c}^a \right] \quad (9-26)$$

Step 4) Re-calculate error to validate feasibility:

After obtaining the updated solution, the error is re-calculated, and the solution with the least error is considered the best solution.

Step 5) Termination:

The process is repeated until maximum iteration. The pseudo-code of the PSSDCA approach is revealed in Algorithm (9-1).

Algorithm (9-1) Pseudo code of developed PSSDCA

```
1:   Input: Solution Initialisation  $\mathcal{C}$ 
2:   Output: Best Solution
3:   begin
4:     Algorithmic parameters are initialised
5:     while stopping criteria are not met, do
6:       for whole solutions
7:         Re-evaluate error using equation (9-14)
8:       Sort agents with error
9:       Discover best solution
10:      Update the SSDCA equation based on (9-15)
11:      Update PO using equation (9-16)
12:      Update proposed PSSDCA using equation
          (9-26)
13:      end for
14:      Validate feasibility of solution using error with
          equation (9-14)
15:      return best solution
16:       $a = a + 1$ 
17:    end while
18:    best solution is attained
19:  end
```

Hence, the output generated using PSSDCA-based DRN is R with dimension $[\tau \times \ell]$.

Prediction of Flight Delay with Developed Gmoa-Based Deeponet

The flight delay happens whenever the flight depart or arrives later than scheduled, resulting in huge economic losses and inconvenience. Precise prediction of flight delays is basic to form the most effective airline business. Recently, several techniques have been adapted for predicting flight delays. Here, the proposed GMOA-based DeepONet is employed for predicting the delay in flight using the fused feature R . The training of DeepONet [62] is performed with the proposed GMOA. The proposed GMOA combines MA [73] and the gradient descent algorithm [74]. The structure of DeepONet and training of DeepONet with proposed GMOA are described in subsections.

a) Structure of DeepONet

DeepONet [62] represents a neural network model for learning non-linear continuous operators, like differential integrals and stochastic Ordinary Differential Equations (ODEs). It is devised based on the universal approximation theorem, which reveals

that a single-layer neural network is adequate for learning a continuous non-linear operator. To minimise the generalised error, DeepONet utilises a branch and trunk module. The training of the branch network is done for encoding the inputted function of the non-linear operator, whereas the trunk network encodes the area of the inputted function. These two layers are combined to produce the output. The structure of DeepONet is shown in Figure 9-3.

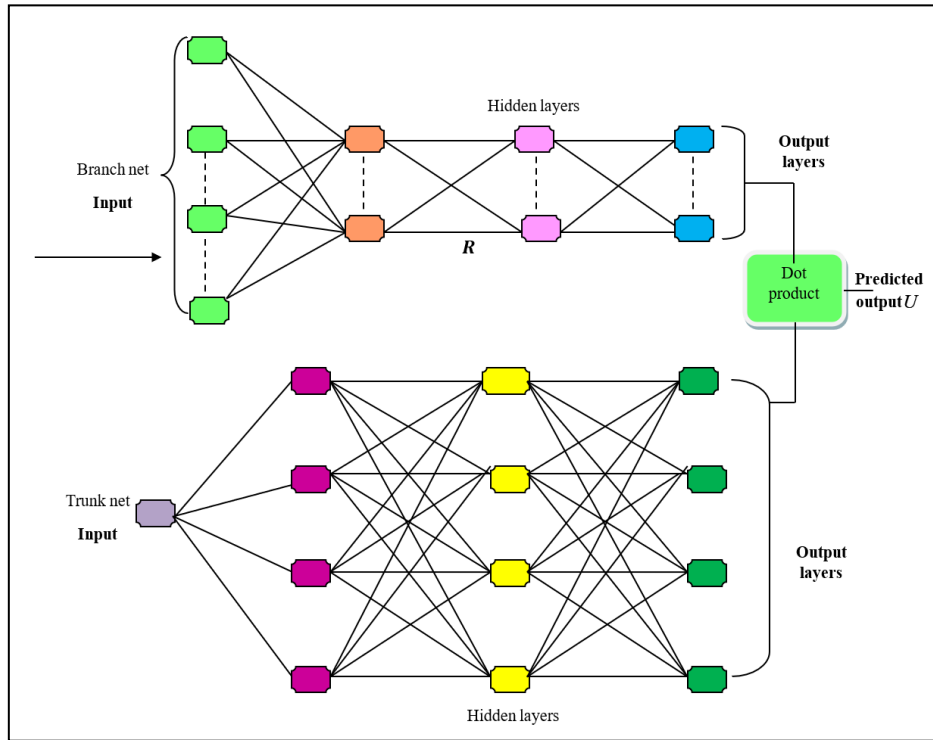


Figure 9-3 Schematic View of DeepONet

Initially, a trunk network is employed that considers i as input and outputs $[n_1, n_2, \dots, n_p]^T \in S^p$. Moreover, in the trunk network, there exists p branch network wherein each acquires $[u(o_1), u(o_2), \dots, u(o_m)]^T$ as input and outputs, a scalar $b_k \in S$ for $k = 1, 2, \dots, p$. Then, the merging of them is done and is given by Equation (9-27):

$$|T(u)(i) - \sum_{k=1}^p \sum_{i=1}^n c_i^k \sigma(\sum_{j=1}^m \zeta_{i,j}^k u(o_j) + \theta_i^{ki}) \sigma(w_k \cdot i + \xi_k)| < \varepsilon \quad (9-27)$$

where $\sigma(\cdot)$ refers to a continuous non-polynomial function, n, p, m are constants, c_i^k , $\zeta_{i,j}^k$, θ_i^{ki} and ξ_k belongs to real number S , $w_k \in S^d$, $o_1 \in K_1$ that is Banach space.

The trunk network adapts activation functions in the last layer such that $t_k = \sigma(\cdot)$ for $k = 1, 2, \dots, p$ and hence this trunk-branch network is considered a trunk network with

each weight linked to the last layer parameterized by other branch networks despite a single variable. Adding bias helps to maximize the performance by minimising the generalization error. The addition of bias $b_0 \in S$ in the last phase, which is given by $T(u)(i) \approx \sum_{k=1}^p b_k t_k + b_0$. The branch networks are merged into one single branch network that results in a vector given by $[b_1, b_2, \dots, b_p]^T \in S_p$. Thus, the output of DeepONet is denoted as U .

b) Training of DeepONet with proposed GMOA

The DeepONet training is performed with developed GMOA and is developed by integrating the Gradient descent algorithm and MA. The MA is motivated by the flight behaviour and mating process in mayflies. The rate of convergence and speed of convergence is superior. The random flight and nuptial dance help to make better trade-offs between the exploitation and exploration phase and helps escape from the local optima. Meanwhile, the Gradient descent algorithm is an effective back-propagation that helps converge faster. Moreover, it can be utilised to discover a provided function's local minimum or maximum. It reduces the error function in an improved way. The method can handle huge datasets. Thus, combining the Gradient descent algorithm and MA helps increase the overall performance rate. The steps of the proposed GMOA are described in the next subsection.

i) Initialisation

The mayfly position in the search space indicates an impending solution. The technique undergoes certain steps. At first, two sets of mayflies are arbitrarily generated, modelling a population of males and females. Here, each mayfly is arbitrarily placed in the e -dimensional vector $W = (W_1, \dots, W_e)$ and its efficiency is computed on specific fitness function $f(W)$. The velocity $V = (V_1, \dots, V_e)$ of a mayfly is described as an alteration of its position and the direction of flying, considering each mayfly is devised based on flying experiences. Here, the personal best position and global best position of mayfly are given as $pbest$ and $gbest$.

ii) Discovery of error

The error of each solution is computed to find the best solution and is given by Equation (9-28):

$$MSE = \frac{1}{F} [\sum_{f=1}^F \varepsilon - U] \quad (9-28)$$

where ε refers to estimated output and U indicates DeepONet output, and F is total data.

iii) Male mayfly's movement

As per MA, the male mayfly position is altered using its own experience and its neighbour's experience. Consider W_a^y represents the present position of a^{th} mayfly on search space at time y and the position is altered by adding velocity V_a^{y+1} to the current position and is expressed as shown in Equation (9-29):

$$W_a^{y+1} = W_a^y + V_a^{y+1} \quad (9-29)$$

The velocity of a male mayfly is given as shown in Equation (9-30):

$$V_a^{y+1} = V_a^y + j_1 e^{-\beta r_p^2} (pbest_a - W_a^y) + j_2 e^{-\beta r_g^2} (gbest_a - W_a^y) \quad (9-30)$$

where, V_a^y signifies velocity of a^{th} mayfly at time y , W_a^y is the current position of mayfly, j_1, j_2 signifies positive attraction constants, $pbest_a$ is the best personal solution of a^{th} mayfly, and $gbest_a$ is the best global solution of a^{th} mayfly.

Here, the personal best solution is given by Equation (9-31):

$$pbest_a = \begin{cases} W_a^{y+1}; & \text{If } f(W_a^{y+1}) < f(pbest_a) \\ kept\ same & \text{Otherwise;} \end{cases} \quad (9-31)$$

The global best solution is given by Equation (9-32):

$$gbest \in \{pbest_1, pbest_2, \dots, pbest_N | f(cbest)\} = \min\{f(pbest_1), f(pbest_2), \dots, f(pbest_N)\} \quad (9-32)$$

where N signifies total mayflies.

The distance is given by Equation (9-33):

$$||w_a - W_a|| = \sqrt{\sum_{v=1}^n (w_{a,v} - W_{a,v})^2} \quad (9-33)$$

where, $w_{a,v}$ indicates v^{th} element of mayfly a and W_a is linked to $pbest_a$ or $gbest$.

The best mayflies keep altering their velocity, which is given as shown by Equation (9-34):

$$V_{a,v}^{y+1} = V_{a,v}^y + e * q \quad (9-34)$$

where e is the nuptial dance coefficient and q is a random number between $[-1, 1]$.

Equation (9-29) can be given as Equation (9-35) to (9-37):

$$W_a^{y+1} = W_a^y + V_a^y + j_1 e^{-\beta r_p^2} (pbest_a - W_a^y) + j_2 e^{-\beta r_g^2} (gbest_a - W_a^y) \quad (9-35)$$

$$W_a^{y+1} = W_a^y + V_a^y + j_1 e^{-\beta r_p^2} pbest_a - j_1 e^{-\beta r_p^2} W_a^y + j_2 e^{-\beta r_g^2} gbest_a - j_2 e^{-\beta r_g^2} W_a^y \quad (9-36)$$

$$W_a^{y+1} = W_a^y [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \quad (9-37)$$

The gradient descent algorithm helps to attain the optimal values quickly. As per the gradient descent algorithm, the equation is given in Equation (9-38):

$$W_a^{y+1} = W_a^y - \vartheta \Delta f(W_a^y) \quad (9-38)$$

where ϑ is a parameter that scales the Gradient as shown by Equation (9-39):

$$W_a^y = \vartheta \Delta f(W_a^y) + W_a^{y+1} \quad (9-39)$$

Substitute Equation (9-39) in Equation (9-37) yields Equations (9-40) to (9-44):

$$W_a^{y+1} = [\vartheta \Delta f(W_a^y) + W_a^{y+1}] [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \quad (9-40)$$

$$W_a^{y+1} = \vartheta \Delta f(W_a^y) [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + W_a^{y+1} [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \quad (9-41)$$

$$W_a^{y+1} - W_a^{y+1} [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] = \vartheta \Delta f(W_a^y) [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \quad (9-42)$$

$$W_a^{y+1} [1 - 1 + j_1 e^{-\beta r_p^2} + j_2 e^{-\beta r_g^2}] = \vartheta \Delta f(W_a^y) [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \quad (9-43)$$

$$W_a^{y+1} [j_1 e^{-\beta r_p^2} + j_2 e^{-\beta r_g^2}] = \vartheta \Delta f(W_a^y) [1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2}] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \quad (9-44)$$

The final updated equation of GMOA is given by Equation (9-45):

$$W_a^{y+1} = \frac{1}{j_1 e^{-\beta r_p^2} + j_2 e^{-\beta r_g^2}} \left[\vartheta \Delta f(W_a^y) \left[1 - j_1 e^{-\beta r_p^2} - j_2 e^{-\beta r_g^2} \right] + V_a^y + j_1 e^{-\beta r_p^2} pbest_a + j_2 e^{-\beta r_g^2} gbest_a \right] \quad (9-45)$$

iv) Female mayfly's movement

The female mayflies do not accumulate in swarms. They fly in the direction of breeding. Consider n_a^y represent the present position of female mayfly a in search space at time y and position is changed by altering the velocity V_a^{y+1} to the current position and is given by Equation (9-46):

$$n_a^{y+1} = n_a^y + V_a^{y+1} \quad (9-46)$$

Based on fitness, the best female attracts the best male, and the second-best female attracts the second-best male. Thus, the velocity can be given as Equation (9-47):

$$V_{a,v}^{y+1} = \begin{cases} V_{a,v}^y + j_2 e^{-\beta r_{mf}^2} (W_{a,v}^y - S_{a,v}^y); & \text{If } f(S_a) > f(W_a) \\ V_{a,v}^y + fl * q; & \text{If } f(S_a) \leq f(W_a) \end{cases} \quad (9-47)$$

where β is the visibility coefficient, r_{mf} is the Cartesian distance among male and female mayflies, fl signifies random walk, and q refers random value between $[-1, 1]$.

v) Mayflies mating

The Crossover operator indicates the process of mating between two mayflies. Considering both female and male populations, a parent mayfly is chosen to make this choice arbitrarily considering fitness. The outcomes of the crossover are given as Equations (9-48) and (9-49):

$$offspring1 = 0 * male + (1 - 0) * female \quad (9-48)$$

$$offspring \leftrightarrow 2 = 0 * female + (1 - 0) * male \quad (9-49)$$

Where *male* signifies the male parent, *female* specifies the female parent, 0 is an arbitrary value. Here, the initial velocity of the offspring is set to 0.

vi) Compute feasibility

The error of each solution is evaluated, and the solution with the least error is selected as the best.

vii) Termination

The steps are repeated till the best solution is generated. Algorithm (9-2) illustrates the pseudo-code of the designed GMOA.

Algorithm 9-2 Pseudo code of designed GMOA

```
1:   Input: Population mayfly  $W$ 
2:   Output:  $W^*$ 
3:   Initialize male and female populations of mayflies
4:   Evaluate error using equation (9-28)
5:   Discover  $g_{best}$  using equation (9-32)
6:   do
7:       while stopping criteria is not satisfied
8:           Update solution of male and female mayflies using
               equations (9-45) and (9-46)
9:           Mate mayflies
10:          Compute offspring using equations (9-48) and (9-49)
11:          Compute feasibility using error with equation (9-28)
12:       end while
13:   end do
```

9.4 Results and Discussion

We perform our experiment on a Personal Computer (PC) with an Intel (R) Core (TM) i7-9700 CPU with a processor speed of 3.00 GHz and 32 GHz RAM. We used libraries such as TensorFlow Core-2.4.1, TensorFlow GPU-2.4.1, NumPy-1.19.1, pandas-0.25.3, sci-kit learn-0.23.2, Scipy1.5.2, PySimpleGUI-4.29.0 and Matplotlib-3.3.1.

9.4.1 Dataset Description

We analysed the downloaded dataset from the US Bureau of Transportation Statistics (BTS) [75]. This dataset comprises flight on-time details with various attributes. The attributes considered in the proposed model include Year, Month, DayofMonth, DayOfWeek, CRSDepTime, CRSArrTime, Tail_Number, Origin, Dest, DepTime, UniqueCarrier, ArrTime, Cancelled, Diverted, FlightNum, UniqueCarrier, Distance, TaxiIn, TaxiOut, ArrDelay and DepDelay. The target attributes considered delays. We used data from January 2021 and November 2021 for the experiment. In January 2021 data, the row size is 361428, and the column size is 21. In November 2021 data, the row size is 547559, and the column size is 21.

9.4.2 Performance Evaluation Measure

We evaluate the performance of the proposed model and other methods using the specific metrics Root Mean Square Error (RMSE), Mean Square Error (MSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE). The metrics are as follows:

a) RMSE

It is the error measure of the square root of the average square difference between actual and predicted and is given by Equation (9-50),

$$RMSE = \sqrt{\frac{\sum_{v=1}^Q (\ell_v - \hat{\ell}_v)^2}{Q}} \quad (9-50)$$

Where v is variable, Q refers to the count of non-missing data, ℓ_v indicates actual data and $\hat{\ell}_v$ refers to estimated data.

b) MSE

It is defined as the average square measure of the difference between the target and estimated class, as shown in Equation (9-28).

c) MAE

Mean Absolute Error is the average of the errors between all the measured values and their true values. The MAE of a prediction can be computed as in Equation (9-51),

$$MAE = \frac{1}{s} \sum_{j=1}^s |x_j - x| \quad (9-51)$$

Where s denotes the total number of samples, $|x_j - x|$ is the absolute error in the prediction.

d) MAPE

Mean Absolute Percent Error is the average of the Absolute Percent Error of the difference between the actual class and predicted class divided by the actual class. The MAPE metric can be estimated using Equation (9-52),

$$MAPE = \frac{1}{n} \sum_{m=1}^n \frac{|x_m - y_m|}{|x_m|} \times 100 \quad (9-52)$$

Where n indicates the total number of samples, x_m is the actual output, and y_m signifies the predicted output.

9.4.3 Comparison of Techniques

We compared all methods with the proposed GMOA-based DeepONet approach for a comprehensive evaluation. The benchmark methods are Deep Belief Network (DBN) [8], Gradient boosting classifier [15], information gain-SVM [64], multi-agent approach [52], and Deep LSTM [76], SSDCA-based Deep LSTM [17] and DeepONet [62].

9.4.4 Comparative Analysis

Our experiment evaluated the GMOA-based DeepONet model with the benchmark methods for flight delay prediction tasks using the evaluation metrics defined in Section 9.4. We perform the analysis using the dataset for January and November 2021, respectively. We further named the methods with special names for simplicity, as shown in Table 9-2.

Table 9-2 Special name for methods

S/No.	Method	Special name
1	DBN	Md1
2	Gradient boosting classifier	Md2
3	Information Gain-SVM	Md3
4	Multi-Agent approach	Md4
5	Deep LSTM	Md5
6	SSDCA-based Deep LSTM	Md6
7	DeepONet	Md7

Assessment using January 2021 data

Table 9-3 displays the experimental results from the evaluation of models on the January 2021 dataset with all the training and testing sets split ratios; the bolded black are the best performance values across all techniques. Figure 9-4 (a-d) shows the performance results plots for all the techniques using the specific evaluation metrics, considering the RMSE metric for evaluation, as seen in Figure 9-4a. For the training and testing set split of 60:40, the performance of all the models, the performance error for four of the benchmark methods are above 0.500, for DBN is 0.906, the Gradient boosting classifier is 0.707, Information Gain-SVM is 0.604, and the Multi-Agent approach is 0.537. While the Deep LSTM is 0.198, SSDCA-based Deep LSTM is 0.197, and DeepONet is 0.191, indicating more improvements of more than 0.200 from the other methods. However, the goal is to have a model with as minimum prediction error as possible because of the high risk associated with a problem like flight delays. Our developed GMOA-based DeepONet further improves by reducing the prediction error to 0.148 out of 1.000. Also, for a dataset split of 70:30, 80:20 and 90:10 training/testing set percentages, the RMSE of our proposed GMOA-based DeepONet

consistently shows improvements better than the baseline methods with values of 0.0143, 0.119 and 0.076 as the best minimal error of predictions.

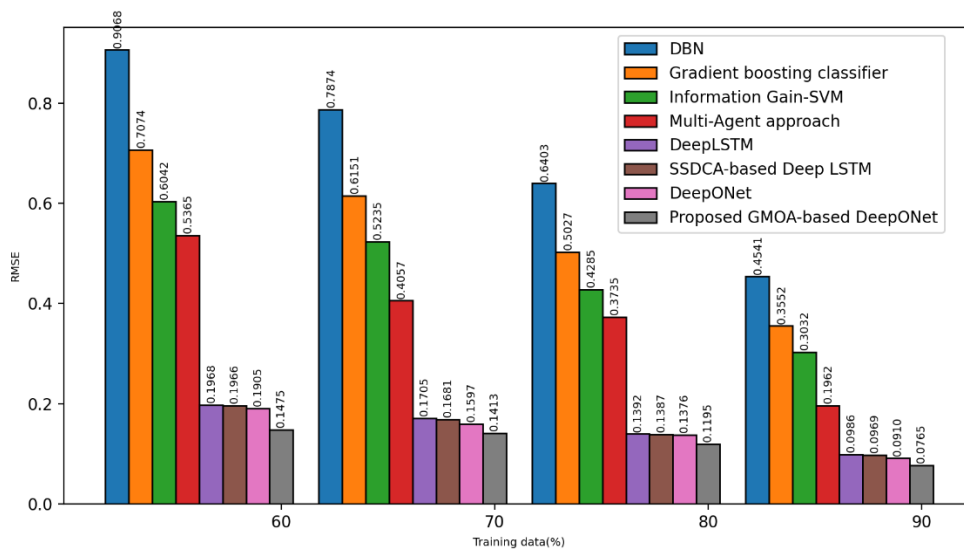
Table 9-3 Results comparison of the proposed method and benchmark methods on various evaluation metric

S/No.	Training/ Testing (%)	Methods/ metrics	Md1	Md2	Md3	Md4	Md5	Md6	Md7	Proposed GMOA- based DeepONet
1	60/40	MSE	0.8223	0.5003	0.3649	0.2878	0.0387	0.0386	0.0363	0.0217
		RMSE	0.9068	0.7073	0.6041	0.5365	0.1968	0.1966	0.1905	0.1475
		MAE	0.5408	0.2805	0.1758	0.1021	0.0386	0.0218	0.0202	0.0194
		MAPE	0.2721	0.1451	0.0947	0.0557	0.0386	0.0218	0.0194	0.0186
2	70/30	MSE	0.6200	0.3783	0.2740	0.1646	0.0291	0.0283	0.02551	0.0199
		RMSE	0.7874	0.6151	0.5235	0.4057	0.1705	0.1681	0.1597	0.1413
		MAE	0.4073	0.2116	0.1329	0.0594	0.0283	0.0199	0.0152	0.0145
		MAPE	0.2046	0.1095	0.0715	0.0331	0.0283	0.0199	0.0145	0.0134
3	80/20	MSE	0.4099	0.2527	0.1836	0.1395	0.0194	0.0192	0.0189	0.0143
		RMSE	0.6403	0.5027	0.4285	0.3735	0.1392	0.1387	0.1376	0.1195
		MAE	0.2709	0.1399	0.0888	0.0492	0.0192	0.0143	0.0099	0.0096
		MAPE	0.1366	0.0723	0.0478	0.0267	0.0019	0.0149	0.0097	0.0096
4	90/10	MSE	0.2062	0.1262	0.0919	0.0385	0.0097	0.0094	0.0083	0.0058
		RMSE	0.4541	0.3552	0.3032	0.1962	0.0986	0.0969	0.0910	0.0765
		MAE	0.1354	0.0704	0.0443	0.0144	0.0094	0.0059	0.0049	0.0049
		MAPE	0.0683	0.0364	0.0238	0.0093	0.0084	0.0059	0.0049	0.0043

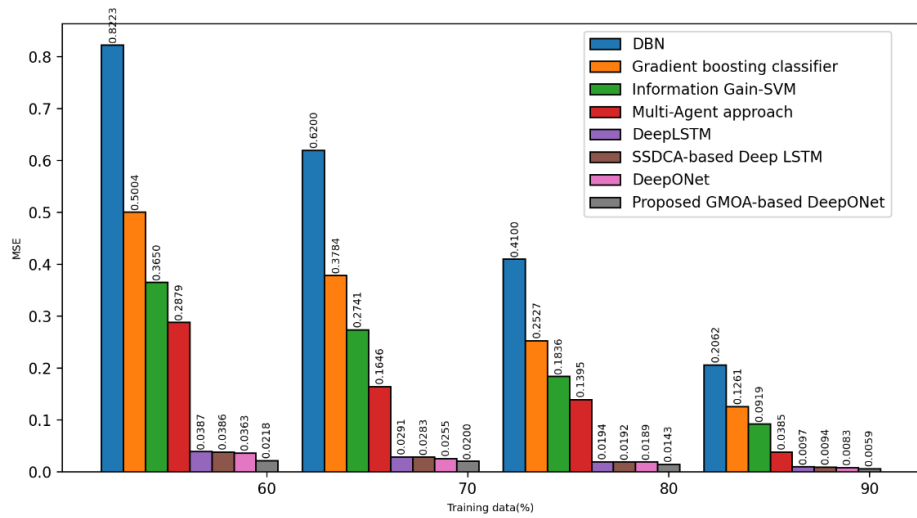
Furthermore, we needed to validate the proposed GMOA-based DeepONet on other performance evaluation metrics because of the differences in computing their errors for vast performance confirmation. We analysed the models with three other metrics mentioned in Section 9.4.3 (MSE, MAE and MAPE, as seen in Figure 9-4b-d). For the dataset split for the 60:40 training and testing set, the proposed GMOA-based DeepONet outperformed the other methods on all evaluation metrics, with the best performance of value of 0.018 on the MAPE. It is worth noticing that some models have the same performance despite the computation methods' differences. For example, Md5 and Md6 on metric MAE and MAPE obtained values of 0.038 and 0.021.

These similarities could be due to the insensitivity of MAE and MAPE to outliers because they are measures for absolute differences in error. It implies that the models learn from the training samples at the same rate as during the training. Also, for dataset splits for 70:30, 80:20 and 90:10 percentages, we observed similarity in Md5 and Md6 on MAE and MAPE metrics. The reason for the similarity may not be far from the sensitivity towards outliers.

The prediction error of the GMOA-based DeepONet reduced consistently in a better pattern than other proposed methods, with best values of 0.0134, 0.009 and 0.004 for the remaining dataset percentage splits. For this study, minimal prediction error represents the rate at which the model correctly predicts flight delays. In commercial aviation, it is important for resource allocation, improving passenger experience, key performance indication and reducing unwanted loss. The consequences of not being aware of the change in flight schedules may result in missing huge contracts by passengers, huge financial loss and poor utilisation of resources. It is much more serious than being delayed at the terminal due to security checks. Therefore, the error rate is an essential indicator in flight delay prediction and analysis. Here, the error varies due to the different computation approaches between actual and predicted. Some take the average of the total sample, while others take only the sum of the difference between the actual and predicted. Considering the four evaluation metrics, we can say that when the datasets are divided into the four training and testing ratio, the overall model effect of the proposed GMOA-based DeepONet is better than the other models. The proposed GMOA-based DeepONet achieved a minimal error of 0.0180 on 60:40 but also continued to reduce the error as the training percentage increased on all metrics.



(a)



(b)

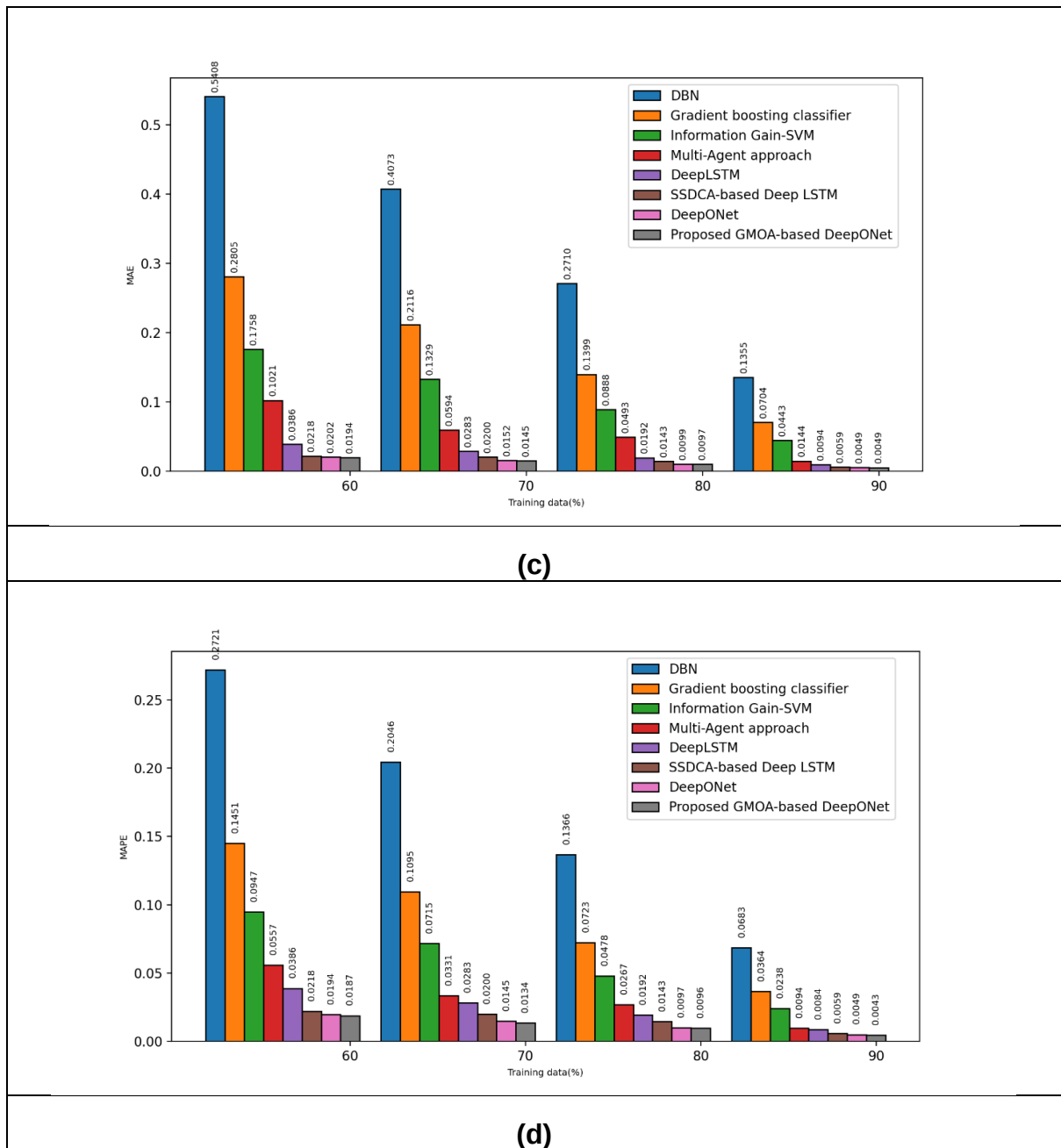


Figure 9-4 Assessment of Techniques using a) RMSE, b) MSE, c) MAE and d) MAPE

Assessment using November 2021 data

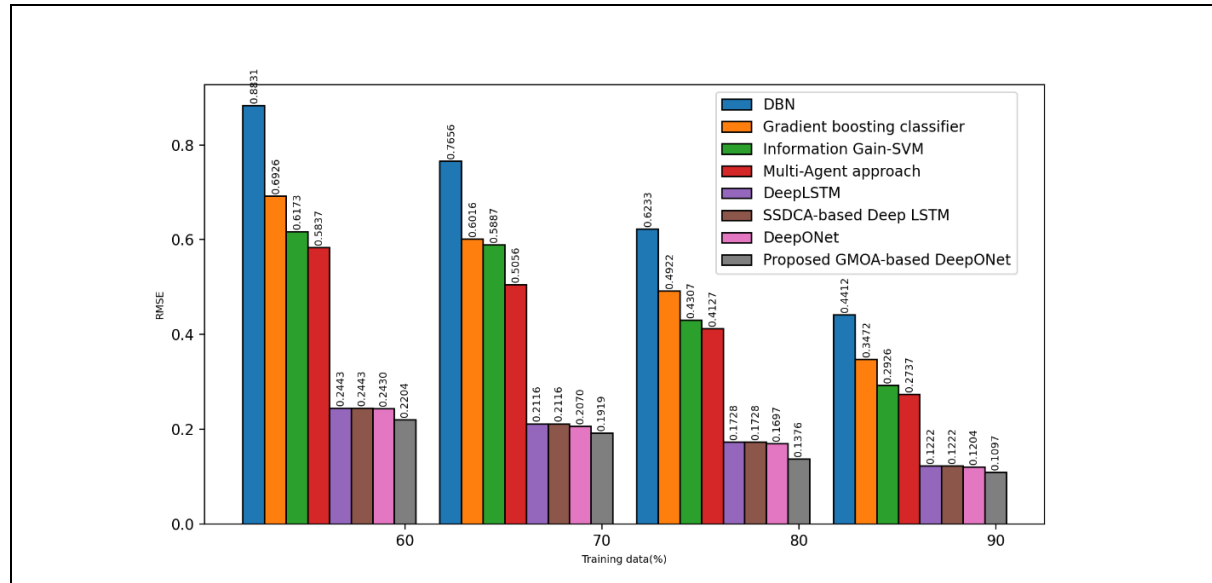
Table 9-4 shows our analysis of techniques on the November 2021 dataset using the metrics: RMSE, MSE, MAE and MAPE, as shown in Figure 9-5(a-d). We evaluated the model's effect with RMSE, as seen in Figure (9-5a). When we split our training and testing set 60:40 per cent, it shows that Md1, Md2, Md3 and Md4 have a high error of more than 0.500 out of 1.000. While Md5, Md6 and Md7 have an error rate below 0.500. However, the Proposed GMOA-based DeepONet outperforms the other methods with a minimal error of 0.220 on RMSE and minimal error of values of 0.048, 0.029 and 0.029 for MSE, MAE and MAPE metrics, respectively, first, regarding the importance of the training/testing splits as observed in Figure 9-5(b-d). Evaluating the models further for 70:40, 80:20, and 90:10 training/testing dataset percentage splits using all the evaluation metrics to measure the performance of the models. However, the other models' error rate shows some decrease. However, the GMOA-based DeepONet still outperforms them with values 0.029, 0.147 and 0.0074 for MAPE on the training/testing splits. It shows that the other models have less result quality from the dataset partition than the developed GMOA-based DeepONet, which consistently shows high differences in decreasing error rates. Indicating that all the predicted flights have higher reliability is significant for flight delay prediction. Therefore, we can conclude that the developed GMOA-based DeepONet model's prediction results are better than the other methods. Our experiment analysis shows that there are no fixed dataset divisions. The dataset train/test partition difference changes the prediction results for different models.

Table 9-4 Results comparison of the proposed method and benchmark methods on various evaluation metric

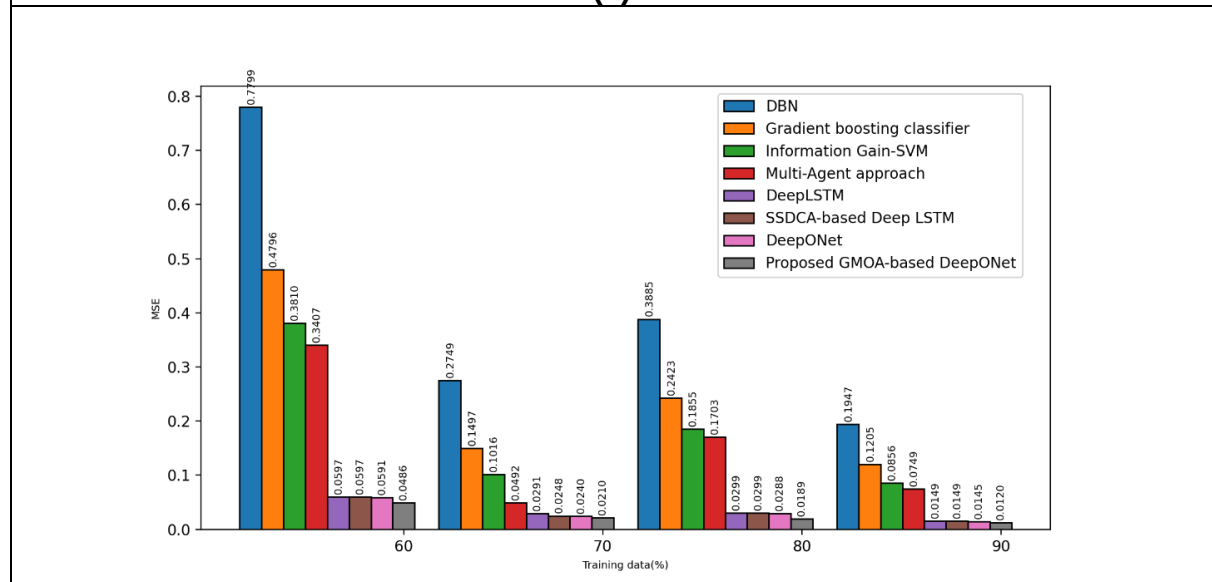
S/No.	Traini ng/ Testi ng (%)	Metho ds/ metrics	Md1	Md2	Md3	Md4	Md5	Md6	Md7	Proposed GMOA- based DeepONet
1	60/40	MSE	0.7799	0.4796	0.3810	0.3407	0.0597	0.0597	0.0591	0.0486
		RMSE	0.8831	0.6926	0.6172	0.5834	0.2443	0.2443	0.2430	0.2204
		MAE	0.5105	0.2699	0.1738	0.1350	0.0597	0.0486	0.0299	0.0298
		MAPE	0.2553	0.1424	0.0975	0.0735	0.0596	0.0486	0.0298	0.0296
2	70/30	MSE	0.2749	0.1497	0.1016	0.0492	0.0291	0.0248	0.0234	0.0209
		RMSE	0.7556	0.6016	0.5887	0.5056	0.2116	0.2116	0.2069	0.1919
		MAE	0.3825	0.2029	0.1311	0.1210	0.0447	0.0368	0.02392	0.0223
		MAPE	0.1915	0.1071	0.0734	0.0646	0.0448	0.0368	0.0223	0.0220
3	80/20	MSE	0.3884	0.2423	0.1855	0.1703	0.0299	0.0298	0.0288	0.0189
		RMSE	0.6232	0.4922	0.4307	0.4127	0.1728	0.1728	0.1697	0.1376
		MAE	0.2549	0.1345	0.0877	0.0661	0.0299	0.0189	0.0155	0.0149
		MAPE	0.1276	0.0710	0.0491	0.0362	0.0299	0.0189	0.0149	0.0147
4	90/10	MSE	0.1947	0.1205	0.0856	0.0749	0.0149	0.0149	0.01448	0.0120
		RMSE	0.4412	0.3471	0.2926	0.2737	0.1222	0.1222	0.1204	0.1097
		MAE	0.1277	0.0674	0.0437	0.0272	0.0149	0.0120	0.0079	0.0075
		MAPE	0.0640	0.0355	0.0244	0.0152	0.0149	0.0120	0.0074	0.0074

We analyse the results from our experiments for the different models using different percentages of the training/test data on the evaluation metrics to explore and exploit the methods employed on the dataset. Our results clearly show that despite similarities in some results of the proposed GMOA-based DeepONet and the DeepONet. Changing the ratio ratios of the splits reveals the importance of having enough training sets because the consistent decrease in the prediction errors of the models is an improvement worth considering when developing critical decision systems such as flight delay decision systems. It is also important to note the high performance of the proposed GMOA-based DeepONet. It shows how important the variable learning rate is from the optimisation algorithm because the consistent improvement in the results against the baseline DeepONet is a result of the variation in the learning rate of the

model as suggested by the optimiser at each iteration. This is also important in the generalisation of the results as it is automated and decided from the global solution returned by the optimiser. Though the best ratio of splits is not known for every model until it is tested from the dataset's knowledge, our experiment shows the best results achieved by the 90%:10% for all the evaluation metrics. This is probably because the higher the knowledge from the historical records for any model to learn, the better the model validation.



(a)



(b)

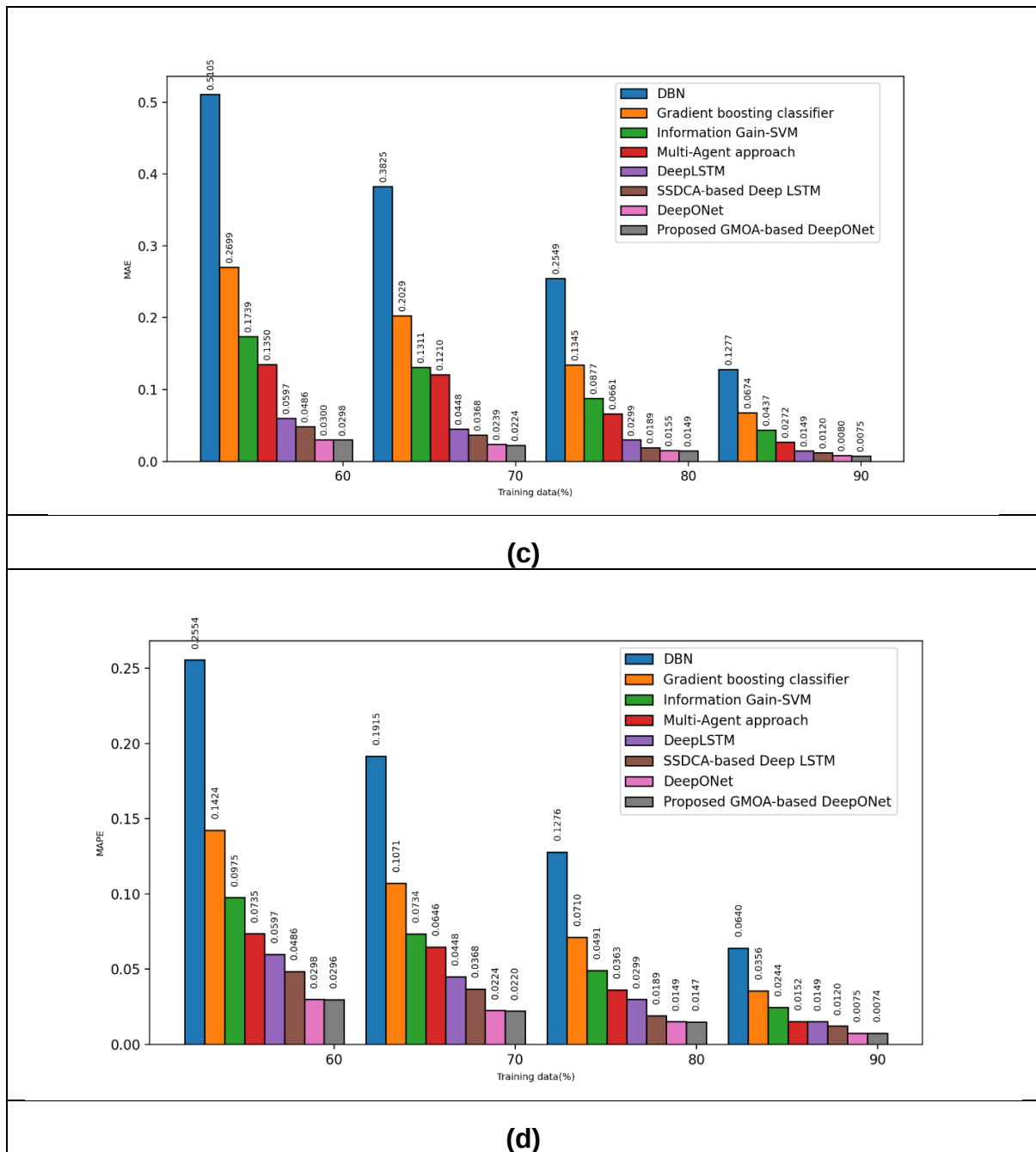


Figure 9-5 Assessment of Techniques using a) RMSE, b) MSE, c) MAE and d) MAPE

Analysis for Delay Prediction

Figure 9-6(a-b) shows the analysis of all techniques for predicting delays for Jan 2021 and Nov 2021. In this research, we are interested in checking our model's efficacy to predict the total number of delays for each day for Jan 2021 and Nov 2021 compared with the ground truth. The prediction of delay with Jan 2021 data is shown in Figure 9-6a). When the date is 15 Jan 2021, the number of flights delayed using the Original, proposed GMOA-based DeepONet, DeepONet, SSDCA-based Deep LSTM, Deep LSTM, Multi-Agent approach, Information Gain-SVM, Gradient boosting

classifier, and DBN are 1254, 1003, 1053, 877, 1239, 877, 2925, 1999 and 2706. When the date is 31 Jan 2021, the number of flights delayed using the Original, proposed GMOA-based DeepONet, DeepONet, SSDCA-based Deep LSTM, Deep LSTM, Multi-Agent approach, Information Gain-SVM, Gradient boosting classifier, and DBN are 2032, 1625, 1808, 1507, 1757, 1422, 1684, 2565 and 3337. The delay assessment with Nov 2021 data is shown in Figure 6b). When the date is 10 Nov 2021, the number of flights delayed using the Original, proposed GMOA-based DeepONet, DeepONet, SSDCA-based Deep LSTM, Deep LSTM, Multi-Agent approach, Information Gain-SVM, Gradient boosting classifier, and DBN are 2086, 1919, 2056, 2223, 3101, 1460, 1460, 2046 and 4105. When the date is 30 Nov 2021, the number of flights delayed using the Original, proposed GMOA-based DeepONet, DeepONet, SSDCA-based Deep LSTM, Deep LSTM, Multi-Agent approach, Information Gain-SVM, Gradient boosting classifier, and DBN are 2568, 2054, 1797, 3447, 2378, 1797, 1797, 2364 and 4563. It reveals the reliability of our proposed methods for the prediction task.

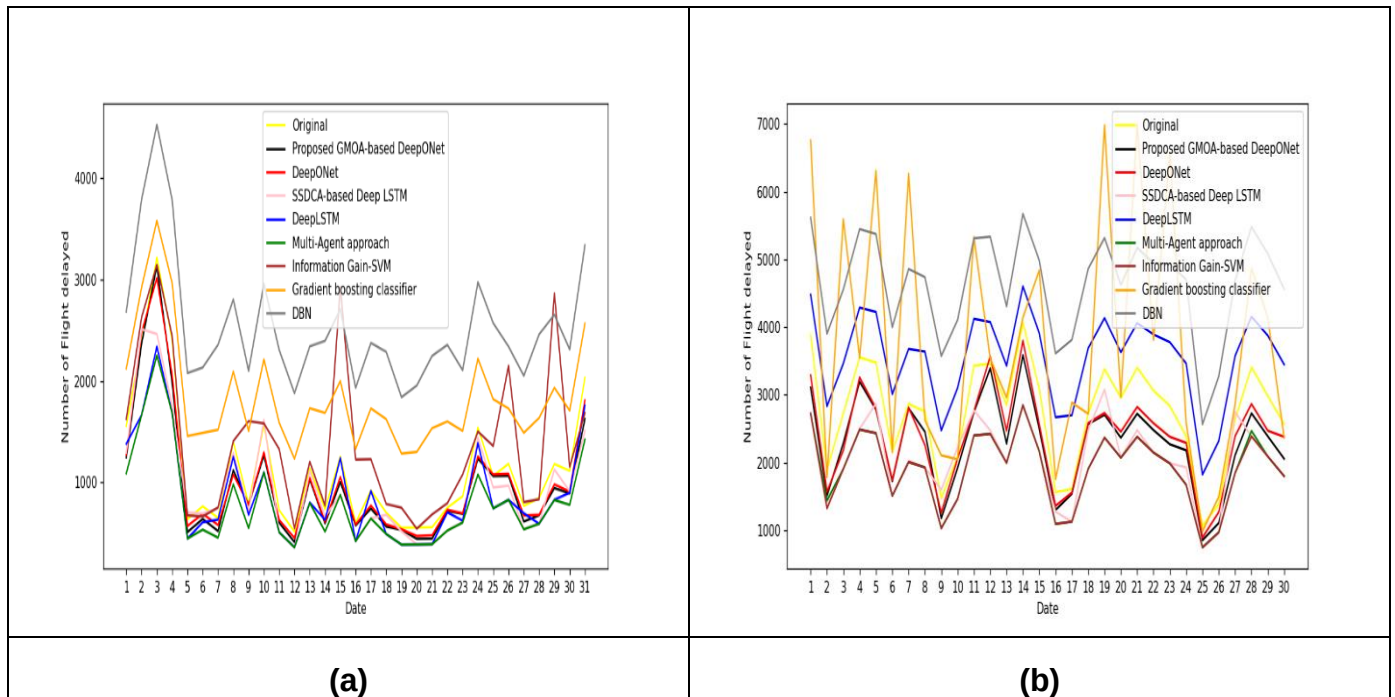


Figure 9-6 Analysis of Techniques using a) Jan 2021 data b) Nov 2021 data

9.4.5 Comparative Discussion of the Techniques

According to the comparative analysis of Table 9-5, the assessment of the proposed GMOA-based DeepONet model established in the paper has the best effect compared to other methods from the literature [15], 4, 32, 44, 48–50], and specific results. As

can be seen from the results, the proposed GMOA-based DeepONet obtained the best minimal RMSE, MSE, MAE and MAPE values of 0.076, 0.005, 0.004 and 0.004 for January 2021 dataset outperforming the other methods. The error results for the proposed GMOA-based DeepONet is considered promising and demonstrate the strong ability of the model prediction to be more reliable. For the November 2021 category, the GMOA-based DeepONet model obtained minimal RMSE, MSE, MAE and MAPE values of 0.109, 0.012, 0.007 and 0.007. But the proposed GMOA-based DeepONet model shows a huge improvement over the traditional DeepONet due to the integration of the optimisation algorithm in finding the learning rate global solution. Generally, the MSE, RMSE, MAE and MAPE are utilised to check how close predictions' estimates are to actual values. Hence, the lower the errors, the closer forecast is to actual values and reveals the method's reliability. It suggests that the generalisation ability of the proposed GMOA-based DeepONet model is stronger than the other methods in our applied datasets. However, there are reasons to believe that the benchmark methods can obtain promising results on other problem application areas or datasets.

Table 9-5: Comparative assessment

<i>Data</i>	<i>Metrics</i>	DBN	Gradient boosting classifier	Informa tion Gain- SVM	Multi- Agent appro ach	Deep LSTM	SSDC A- based Deep LSTM	DeepO Net	Developed GMOA- based DeepONet
<i>Jan. 2021</i>	<i>RMSE</i>	0.4541	0.3551	0.3032	0.1962	0.0986	0.0969	0.0910	0.0765
	<i>MSE</i>	0.2062	0.1262	0.0919	0.0385	0.0097	0.0093	0.0083	0.0059
	<i>MAE</i>	0.1354	0.0704	0.0443	0.0144	0.0094	0.0059	0.0049	0.0048
	<i>MAPE</i>	0.0683	0.0364	0.0238	0.0093	0.0084	0.0059	0.0049	0.0043
<i>Nov. 2021</i>	<i>RMSE</i>	0.4412	0.3472	0.2926	0.2737	0.1222	0.1222	0.1204	0.1097
	<i>MSE</i>	0.1947	0.1205	0.0856	0.0749	0.0159	0.0149	0.0145	0.0120
	<i>MAE</i>	0.1277	0.0674	0.0436	0.0272	0.0149	0.0120	0.0079	0.0074
	<i>MAPE</i>	0.0640	0.0356	0.0244	0.0152	0.0149	0.0120	0.0074	0.0074

9.5 Conclusion and Lesson Learned

It is critical to estimate the flight delay accurately and promptly in an intelligent aviation system. Predicting flight delays can give travellers dependable travel plans to reduce anxiety and give airports and airlines more proactive operation methods to improve service quality. This research uses flight information on multiple air route routes for flight delay prediction. We introduced a novel Deep Learning approach by extending a state-of-the-art method known as DeepONet for predicting flight delays by integrating the mayfly optimisation algorithm. According to our limited knowledge, it is the first attempt at combining the DeepONet with GMOA. It is part of an ongoing research project investigating improving air passenger experience through reliable flight delay predictions based on Deep Learning models. We studied the efficacy of integrating DeepONet and GMOA for flight delay prediction tasks. Our results show a good fit and can achieve minimal prediction error compared with other benchmark techniques.

Moreover, we have considered only the non-weather delays at all the available airports. The Box-Cox transformation normalises the distributed data and eliminates potential collinearity among variables. The modified DRN is employed to perform the feature fusion. DRN's weight and bias update is done using PSSDCA, which is obtained by integrating SSD, CAViaR and PO. For the performance evaluation of the proposed GMOA-based DeepONet model, we utilise real-world flight data on-time from the US Transportation Bureau of Statistics in the numerical study. The new training algorithm devised for predicting the flight delay uses the DeepONet by tuning the optimal weights. Here, the weight update of DeepONet is performed with the GMOA. Compared with the other benchmark techniques, the GMOA-based DeONet model outperforms the other models. Despite the air transportation sophistication in the empirical, the GMOA-based DeepONet performs well in the flight delay prediction task with low MSE, RMSE, MAE, and MAPE. Specifically, the proposed GMOA-based DeepONet provided superior performance with the smallest RMSE of 0.085, smallest MSE of 0.007, smallest MAE of 0.005, and smallest MAPE of 0.007. The result shows that our novel GMOA proposed in this research is vital in improving flight delay prediction accuracy.

Our research contributes to the global discussions in the literature on efficient flight delay prediction models, and there are some possible areas of extension. Further

openness and analysis of these trustworthy flight plans are essential for the airline industry's performance as air passengers want more information and dependability in their travel itineraries. In the future, we recommend other nature-inspired optimisation techniques and databases to check the proposed approach's reliability. Furthermore, in the data science community, the Physics-Informed-Machine-Learning (PIML) rapidly emerges, which seeks to develop a research discipline that leverages domain-specific or wealth of science to improve the effectiveness of the Deep Learning model. The main aim of PIML is to advance scientific understanding by generalising Machine Learning models and ensuring they are consistent and interpretable. Therefore, in the future, we will also attempt to use PIML methods for model development for flight delay prediction. In addition, high-performance computing, parallel computing and cloud computing facilities widen the research scope with real-time prediction results and reduce computation time.

Acknowledgement and Funding

The Petroleum Trust Development Fund (PTDF) Nigeria partially funded this work through grant number PTDF/ED/OSS/PHD/DBB/1558/19 for one of the first author's PhD studies. We also thank the UKRI for the Covid-19 recovery grant under the budget code SA077N.

Data Availability

All data used in this research is available at <https://www.bts.gov/>.

9.6 References

- [1] Z. Guo *et al.*, "Sgdan—A spatio-temporal graph dual-attention neural network for quantified flight delay prediction," *Sensors (Switzerland)*, vol. 20, no. 22, pp. 1–18, 2020, doi: 10.3390/s20226433.
- [2] B. Bubalo and J. R. Daduna, "Airport capacity and demand calculations by simulation-the case of Berlin-Brandenburg International Airport," *NETNOMICS Econ. Res. Electron. Netw.*, vol. 12, no. 3, pp. 161–181, 2011, doi: 10.1007/s11066-011-9065-6.
- [3] R. Wang, Shu-Li, Hsu, Y. H. Lin, and M.-L. Tseng, "Evaluation of customer perceptions on airline service quality in uncertainty," *Procedia - Soc. Behav. Sci.*, vol. 25, no. 2011, pp. 419–437, 2011, doi: 10.1016/j.sbspro.2012.02.054.
- [4] B. Zou and M. Hansen, "Flight delays, capacity investment and social welfare under air transport supply-demand equilibrium," *Transp. Res. Part A Policy Pract.*, vol. 46, no. 6, pp. 965–980, 2012, doi: 10.1016/j.tra.2012.02.015.
- [5] Y. He, M. Hu, L. Yuan, and H. Jiang, "Flight departure time prediction based on Deep Learning," p. 23, 2022, doi: 10.1117/12.2623856.
- [6] J. Chen and M. Li, "Chained predictions of flight delay using Machine Learning," in *AIAA Scitech 2019 Forum*, 2019, no. January, pp. 1–25. doi: 10.2514/6.2019-1661.
- [7] C. Yang, Z. A. Marshall, and J. H. Mott, "A novel integration platform to reduce flight delays in the National Airspace System," *2020 Syst. Inf. Eng. Des. Symp. SIEDS 2020*, pp. 1–6, 2020, doi: 10.1109/SIEDS49339.2020.9106657.
- [8] B. Yu, Z. Guo, S. Asian, H. Wang, and G. Chen, "Flight delay prediction for commercial air transport: A Deep Learning approach," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 125, no. March, pp. 203–221, 2019, doi: 10.1016/j.tre.2019.03.013.
- [9] S. Lan, "Planning for robust airline operations: optimizing aircraft routings and flight departure times to achieve minimum passenger disruptions," 2003. [Online]. Available: <https://dspace.mit.edu/bitstream/handle/1721.1/17568/52769892-MIT.pdf?sequence=2&isAllowed=y>
- [10] R. A. Sugara and D. Purwitasari, "Flight Delay Prediction for Mitigation of Airport Commercial Revenue Losses Using Machine Learning on Imbalanced Dataset," no. November, pp. 6–13, 2022.
- [11] S. Mokhtarimousavi and A. Mehrabi, "Flight delay causality: Machine Learning technique in conjunction with random parameter statistical analysis," *Int. J. Transp. Sci. Technol.*, vol. 12, no. 1, pp. 230–244, 2022, doi: 10.1016/j.ijtst.2022.01.007.
- [12] C. Liu *et al.*, "Airport flight ground service time prediction with missing data using graph convolutional neural network imputation and bidirectional sliding mechanism," *Appl. Soft Comput.*, vol. 133, p. 109941, 2023, doi: 10.1016/j.asoc.2022.109941.
- [13] Y. Cao, C. Zhu, Y. Wang, and Q. Li, "A Method of Reducing Flight Delay by Exploring Internal Mechanism of Flight Delays," *J. Adv. Transp.*, vol. 2019, no. 5, pp. 22–25, 2019, doi: 10.1155/2019/7069380.

- [14] D. B. Bisandu, M. S. Homaid, I. Moulitsas, and S. Filippone, "A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective," in *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK*, 2021, pp. 1–10. doi: <https://doi.org/10.1145/3505711.3505712>.
- [15] N. Chakrabarty, "A data mining approach to flight arrival delay prediction for American airlines," in *IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference*, 2019, pp. 102–107. doi: 10.1109/IEMECONX.2019.8876970.
- [16] S. Y. Ahmed, "Design of mobile application for travelers to transport baggage and handle Check-in process," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 2, pp. 28–33, 2018, doi: 10.14569/IJACSA.2018.090205.
- [17] D. B. Bisandu, I. Moulitsas, and S. Filippone, "Social ski driver conditional autoregressive-based Deep Learning classifier for flight delay prediction," *Neural Comput. Appl.*, vol. 34, no. 11, pp. 8777–8802, 2022, doi: 10.1007/s00521-022-06898-y.
- [18] C. Manasa and P. Velayutham, "Predicting flight delays with error calculation using machine learned classifiers," *J. Eng. Sci.*, vol. 14, no. 01, pp. 93–97, 2023, doi: 10.1109/SPIN48934.2020.9071159.
- [19] R. Khan, S. Akbar, and T. A. Zahed, "Flight Delay Prediction Based on Gradient Boosting Ensemble Techniques," pp. 1–5, 2023, doi: 10.1109/icosst57195.2022.10016828.
- [20] Faiza and K. Khalil, "Airline flight delays using Artificial Intelligence in COVID-19 with perspective analytics," *J. Intell. Fuzzy Syst.*, pp. 1–23, 2023, doi: 10.3233/jifs-222827.
- [21] M. D. PeCoy and M. A. Redmond, "Flight reliability during periods of high uncertainty," *J. Air Transp. Manag.*, vol. 106, no. August 2022, 2023, doi: 10.1016/j.jairtraman.2022.102322.
- [22] S. K. Fatima *et al.*, "A Comprehensive Framework For Model Prediction of Aerial Vehicles: Perspective Analysis and Comparison," no. January, 2023.
- [23] S. Chen, B. Chen, P. Shu, Z. Wang, and C. Chen, "Real-time unmanned aerial vehicle flight path prediction using a bi-directional long short-term memory network with error compensation," *J. Comput. Des. Eng.*, vol. 10, no. 1, pp. 16–35, 2023, doi: 10.1093/jcde/qwac125.
- [24] Z. Tang, S. Huang, X. Zhu, W. Pan, S. Han, and T. Gong, "Research on the multilayer structure of flight delay in China air traffic network," *Phys. A Stat. Mech. its Appl.*, vol. 609, p. 128309, 2023, doi: 10.1016/j.physa.2022.128309.
- [25] A. Abdelghany, V. S. Guzhva, and K. Abdelghany, "The limitation of machine-learning based models in predicting airline flight block time," *J. Air Transp. Manag.*, vol. 107, no. November 2021, p. 102339, 2023, doi: 10.1016/j.jairtraman.2022.102339.
- [26] A. Khanal, R. Bhusal, K. Subbarao, A. Chakravarthy, and W. Okolo, "Learning Stochastic Processes Using Gaussian Processes: An Application to Flight Delay Prediction," no. January, pp. 1–13, 2023, doi: 10.2514/6.2023-1257.

- [27] J. Qu, S. Wu, and J. Zhang, "Flight Delay Propagation Prediction Based on Deep Learning," *Mathematics*, vol. 11, no. 3, p. 494, 2023, doi: 10.3390/math11030494.
- [28] X. Ma, Z. He, P. Yang, X. Liao, and W. Liu, "Agent-based modelling and simulation for life-cycle airport flight planning and scheduling," *J. Simul.*, vol. 00, no. 00, pp. 1–14, 2023, doi: 10.1080/17477778.2023.2169643.
- [29] M. F. Yazdi, S. R. Kamel, S. J. M. Chabok, and M. Kheirabadi, "Flight delay prediction based on Deep Learning and Levenberg-Marquart algorithm," *J. Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00380-z.
- [30] Y. Ding, "Predicting flight delay based on multiple linear regression," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 81, no. 1, pp. 0–7, 2017, doi: 10.1088/1755-1315/81/1/012198.
- [31] J. Wang, X. Liu, and J. Ding, "Air passenger travel forecasting model based on both dynamical individual behavior and social influence force," *J. Algorithms Comput. Technol.*, vol. 13, pp. 1–9, 2019, doi: 10.1177/1748302619881392.
- [32] M. Yap, D. Luo, O. Cats, N. van Oort, and S. Hoogendoorn, "Where shall we sync? Clustering passenger flows to identify urban public transport hubs and their key synchronization priorities," *Transp. Res. Part C Emerg. Technol.*, vol. 98, no. December 2018, pp. 433–448, 2019, doi: 10.1016/j.trc.2018.12.013.
- [33] R. Boutaba *et al.*, "Comprehensive survey Machine Learning," *J. of Internet Serv. and Applications*, vol. 9, no. 16, p. 99, 2018, doi: <https://doi.org/10.1186/s13174-018-0087-2>.
- [34] S. Ayoubi *et al.*, "Machine Learning for Cognitive Network Management," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 158–165, 2018, doi: 10.1109/MCOM.2018.1700560.
- [35] K. Q. Cai, J. Zhang, M. M. Xiao, K. Tang, and W. B. Du, "Simultaneous optimisation of airspace congestion and flight delay in air traffic network flow management," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 11, pp. 3072–3082, 2017, doi: 10.1109/TITS.2017.2673247.
- [36] K. Cai, Y. Jia, Y. Zhu, and M. Xiao, "A novel biobjective risk-based model for stochastic air traffic network flow optimisation problem," *Sci. World J.*, vol. 2015, 2015, doi: 10.1155/2015/742541.
- [37] J. Chen, H. Huang, S. Tian, and Y. Qu, "Feature selection for text classification with Naïve Bayes," *Expert Syst. Appl.*, vol. 36, no. 3 PART 1, pp. 5432–5435, 2009, doi: 10.1016/j.eswa.2008.06.054.
- [38] T. Zhang, D. Tao, X. Qu, X. Zhang, R. Lin, and W. Zhang, "The roles of initial trust and perceived risk in public's acceptance of automated vehicles," *Transp. Res. Part C Emerg. Technol.*, vol. 98, no. November 2018, pp. 207–220, 2019, doi: 10.1016/j.trc.2018.11.018.
- [39] J. Qu, T. Zhao, M. Ye, J. Li, and C. Liu, "Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data," *Neural Process. Lett.*, vol. 52, no. 2, pp. 1461–1484, 2020, doi: 10.1007/s11063-020-10318-4.
- [40] and M. C. Liu, Xia, Fang Wan, Lei Chen, Zhao Qiu, *Research on traffic passenger volume prediction of Sanya city based on ARIMA and grey markov models*, no. 2(153). Springer, Singapore, 2018. doi:

- 10.30748/soi.2018.153.08.
- [41] J. Huo, K. L. Keung, C. K. M. Lee, K. K. H. Ng, and K. C. Li, "The prediction of flight delay: Data-driven Machine Learning approach," *IEEE Int. Conf. Ind. Eng. Eng. Manag.*, vol. 2020-Decem, pp. 190–194, 2020, doi: 10.1109/IEEM45057.2020.9309919.
 - [42] M. Zoutendijk and M. Mitici, "Probabilistic flight delay predictions using Machine Learning and applications to the flight-to-gate assignment problem," *Aerospace*, vol. 8, no. 6, p. 152, 2021, doi: 10.3390/aerospace8060152.
 - [43] Dutrieux S. C. M., "Predicting flight delay distributions: A Machine Learning-based approach at a regional airport," the Delft University of Technology, 2021. [Online]. Available: <https://repository.tudelft.nl/islandora/object/uuid:8d682d8f-0e05-4170-8a8d-406bb3854f99>
 - [44] M. Ben Messaoud, "A thorough review of aircraft landing operation from practical and theoretical standpoints at an airport which may include a single or multiple runways," *Appl. Soft Comput.*, vol. 98, p. 106853, 2021, doi: 10.1016/j.asoc.2020.106853.
 - [45] D. Karádi, C. Csiszár, and E. Nagy, "Integrated information application on mobile devices for air passengers," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2015*, 2015, pp. 304–311. doi: 10.1109/MTITS.2015.7223272.
 - [46] H. Jiang and Y. Zhang, "An assessment of passenger experience at Melbourne Airport," *J. Air Transp. Manag.*, vol. 54, pp. 88–92, 2016, doi: 10.1016/j.jairtraman.2016.04.002.
 - [47] S. Zhu and Y. Wang, "Scaled sequential threshold least-squares (S² TLS) algorithm for sparse regression modelling and flight load prediction," *Aerosp. Sci. Technol.*, vol. 85, pp. 514–528, 2019, doi: 10.1016/j.ast.2018.12.038.
 - [48] C. Park *et al.*, "ST-GRAT: A Novel Spatio-temporal Graph Attention Networks for Accurately Forecasting Dynamically Changing Road Speed," *Int. Conf. Inf. Knowl. Manag. Proc.*, pp. 1215–1224, 2020, doi: 10.1145/3340531.3411940.
 - [49] R. Yu, Y. Li, C. Shahabi, U. Demiryurek, and Y. Liu, "Deep Learning: A generic approach for extreme condition traffic forecasting," *Proc. 17th SIAM Int. Conf. Data Mining, SDM 2017*, pp. 777–785, 2017, doi: 10.1137/1.9781611974973.87.
 - [50] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, "Urban traffic prediction from spatio-temporal data using deep meta learning," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 1, pp. 1720–1730, 2019, doi: 10.1145/3292500.3330884.
 - [51] J. Zhang, X. Shi, J. Xie, H. Ma, I. King, and D. Y. Yeung, "GaAN: Gated attention networks for learning on large and spatiotemporal graphs," *34th Conf. Uncertain. Artif. Intell. 2018, UAI 2018*, vol. 1, pp. 339–349, 2018.
 - [52] Y. Guleria, Q. Cai, S. Alam, and L. Li, "A Multi-Agent Approach for Reactionary Delay Prediction of Flights," *IEEE Access*, vol. 7, no. Cdm, pp. 181565–181579, 2019, doi: 10.1109/ACCESS.2019.2957874.
 - [53] G. Antoniou *et al.*, "Enabling the use of a planning agent for urban traffic

- management via enriched and integrated urban data,” *Transp. Res. Part C Emerg. Technol.*, vol. 98, no. July 2018, pp. 284–297, 2019, doi: 10.1016/j.trc.2018.12.005.
- [54] S. Hou, Z. Gong, Q. Nie, Q. Xiao, and Y. Tan, “A hybrid predictive model for high-frequency and multi-periodic data in call center of online travel agency,” *Proc. - 2018 IEEE Int. Conf. Data, Data 2018*, vol. d, pp. 3128–3134, 2019, doi: 10.1109/BigData.2018.8621977.
 - [55] J. Drchal, M. Čertický, and M. Jakob, “Data-driven activity scheduler for agent-based mobility models,” *Transp. Res. Part C Emerg. Technol.*, vol. 98, no. December 2018, pp. 370–390, 2019, doi: 10.1016/j.trc.2018.12.002.
 - [56] M. N. Postorino and L. Mantecchini, “An Agent Framework to Support Air Passengers in Departure Terminals Study and Optimisation of Human Organ Transportation Network for Transplant Activities View project Research on Air Transport and Airport Industry Market Trends View project An Agent Fram,” 2018.
 - [57] L. Cheg, C. Fookes, V. Reddy, and P. Yarlagaadda, “Analysis of passenger group behaviour and its impact on passenger flow using an agent-based model,” *2014 4th Int. Conf. Simul. Model. Methodol. Technol. Appl.*, pp. 733–738, 2014, doi: 10.5220/0005086807330738.
 - [58] European Union Aviation Safety Agency, “Artificial Intelligence Roadmap: A Human-Centric Approach to AI in Aviation,” no. February, pp. 1–33, 2020, [Online]. Available: <https://www.easa.europa.eu/newsroom-and-events/news/easa-artificial-intelligence-roadmap-10-published>
 - [59] S. Somani, P. Priyanshu, S. Meghna, and S. M, “An approach of applying Machine Learning model in flight delay prediction- A comparative analysis,” *Rev. Gestão Inovação e Tecnol.*, vol. 11, no. 3, pp. 1233–1244, 2021, doi: 10.47059/revistageintec.v11i3.2006.
 - [60] W. Wu, K. Cai, Y. Yan, and Y. Li, “An Improved SVM Model for Flight Delay Prediction,” *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, vol. 2019-Septe, 2019, doi: 10.1109/DASC43569.2019.9081611.
 - [61] C. H. Yeh and Y. L. Kuo, “Evaluating passenger services of Asia-Pacific international airports,” *Transp. Res. Part E Logist. Transp. Rev.*, vol. 39, no. 1, pp. 35–48, 2003, doi: 10.1016/S1366-5545(02)00017-0.
 - [62] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, “Learning non-linear operators via DeepONet based on the universal approximation theorem of operators,” vol. 3, no. March, pp. 31–33, 2021, doi: 10.1038/s42256-021-00302-5.
 - [63] M. Guvercin, N. Ferhatosmanoglu, and B. Gedik, “Forecasting Flight Delays Using Clustered Models Based on Airport Networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 5, pp. 3179–3189, 2021, doi: 10.1109/TITS.2020.2990960.
 - [64] Z. Chen, P. Wanke, J. J. M. Antunes, and N. Zhang, “Chinese airline efficiency under CO2 emissions and flight delays: A stochastic network DEA model,” *Energy Econ.*, vol. 68, pp. 89–108, 2017, doi: 10.1016/j.eneco.2017.09.015.
 - [65] J. Yi, H. Zhang, H. Liu, G. Zhong, and G. Li, “Flight Delay Classification Prediction Based on Stacking Algorithm,” *J. Adv. Transp.*, vol. 2021, 2021,

- doi: 10.1155/2021/4292778.
- [66] Z. Guo, B. Yu, M. Hao, W. Wang, Y. Jiang, and F. Zong, "A novel hybrid method for flight departure delay prediction using Random Forest Regression and Maximal Information Coefficient," *Aerosp. Sci. Technol.*, vol. 116, p. 106822, 2021, doi: 10.1016/j.ast.2021.106822.
 - [67] W. Shao, A. Prabowo, S. Zhao, P. Koniusz, and F. D. Salim, "Predicting flight delay with spatio-temporal trajectory convolutional network and airport situational awareness map," *Neurocomputing*, vol. 472, pp. 280–293, 2022, doi: 10.1016/j.neucom.2021.04.136.
 - [68] R. Maclejewski, A. Pattath, S. Ko, R. Hafen, W. S. Cleveland, and D. S. Ebert, "Automated box-cox transformations for improved visual encoding," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 1, pp. 130–140, 2013, doi: 10.1109/TVCG.2012.64.
 - [69] Z. Chen, Y. Chen, L. Wu, S. Cheng, and P. Lin, "Deep residual network based fault detection and diagnosis of photovoltaic arrays using current-voltage curves and ambient conditions," *Energy Convers. Manag.*, vol. 198, no. May, p. 111793, 2019, doi: 10.1016/j.enconman.2019.111793.
 - [70] Q. Askari, I. Younas, and M. Saeed, "Political Optimizer: A novel socio-inspired meta-heuristic for global optimisation," *Knowledge-Based Syst.*, vol. 195, p. 105709, 2020, doi: 10.1016/j.knosys.2020.105709.
 - [71] A. Tharwat and T. Gabel, "Parameters optimisation of support vector machines for imbalanced data using social ski driver algorithm," *Neural Comput. Appl.*, vol. 32, no. 11, pp. 6925–6938, 2020, doi: 10.1007/s00521-019-04159-z.
 - [72] R. F. Engle and S. Manganelli, "CAViaR: Conditional autoregressive value at risk by regression quantiles," *J. Bus. Econ. Stat.*, vol. 22, no. 4, pp. 367–381, 2004, doi: 10.1198/073500104000000370.
 - [73] K. Zervoudakis and S. Tsafarakis, "A mayfly optimisation algorithm," *Comput. Ind. Eng.*, vol. 145, no. September 2019, p. 106559, 2020, doi: 10.1016/j.cie.2020.106559.
 - [74] S. Wang, H. Wang, and P. Perdikaris, "Improved architectures and training algorithms for deep operator networks," pp. 1–40, 2021, [Online]. Available: <http://arxiv.org/abs/2110.01654>.
 - [75] United States Department of Transportation, "Bureau of Transportation Statistics, Bureau of Transportation Statistics," *United States Department of Transportation*, 2020. <https://www.bts.gov/> (accessed Mar. 03, 2020).
 - [76] S. Kim, S. Hong, M. Joh, and S. K. Song, "DeepRain: ConvLSTM network for precipitation prediction using multichannel radar data," *arXiv*, pp. 3–6, 2017.

Chapter 10: General Discussion and Findings

This Chapter presents a general discussion of the methods and findings in this thesis. The methods focus mainly on classifying and predicting airline flight delays and can serve as models for developing smart decision support systems. They are based on the ground truth data of on-time performance from the United States Bureau of Transportation Statistics. Predicting flight delays in a complex network, such as the air transportation system, is a challenging and daunting task due to the complexity of the system processes and the nature of the causal factors. Identifying abnormal patterns in a large amount of the generated ground truth dataset is highly difficult because of the non-linear relationships among the processes, components and subsystems. There is difficulty in accurately and reliably classifying or predicting flight delays in air transport systems. The different causal factors, intrinsic data characteristics, distributions and class overlaps between arrival and departure flights, and irregular patterns of occurrences of delays may impact the performances of the learning algorithms. In the aviation industry, reliable flight delay prediction is a key performance indicator because of the economic impact on daily airline business. As reported by the airlines and other stakeholders of the aviation industry, a single flight delay of at least 15 minutes has a high negative impact on the business processes and passenger experiences, costing a huge amount of money [1–8]. Therefore, reducing the percentage of flight delays can significantly benefit airlines, airports and passengers. The dataset employed in this thesis has two target classes: Not delayed (0) and delayed (1). These target classes are based on the ground truth and contain the original overall BTS dataset. Hence, different approaches for the classification and prediction of flight delays, handling factors, complex relationships and nature are proposed and implemented with validation based on the ground truth classification and prediction by the developed models through benchmark evaluation metrics.

To address all the potential research gaps this thesis identifies, mentioned in Chapter 3. Algorithms and strategies have been proposed, implemented and validated. These algorithms and approaches are explained as follows:

10.1 A Deep Feedforward Neural Network Algorithm for Delay Prediction and Analysis

The flight delay on-time performance dataset holds valuable hidden patterns essential for data analysis, such as predictive analysis and pattern recognition. However, the complexity of relationships among various factors makes uncovering these patterns challenging due to non-linearity. To address this, we propose an approach for flight delay predictive modelling, utilising filter-based feature extraction to select relevant features correlated with the target class. This approach combines deep feedforward neural networks with filtering techniques for pre-processing and training the model. Results demonstrate that our methodology outperforms similar approaches in the literature, providing at least 20% improvement in classifying delayed and non-delayed flights. The integrated feature filtering techniques and deep neural network contribute to this performance boost. The impact of our algorithm is significant as flight delays cause economic losses and affect passenger experiences. Accurate predictions can lead to improved planning, better passenger experience, and increased non-aviation income through retail shopping at airports. While our approach shows good results, we acknowledge the need for further optimisation to reduce false-negative and false-positive rates and minimize potential errors.

10.2 An Enhanced Stacked Autoencoder Algorithm for Flight Delay Prediction

This Chapter addresses fundamental questions regarding the modification of stack-autoencoder architectures to compare their performances in flight prediction tasks. To answer this, we developed an enhanced deep autoencoder algorithm with three different architectures to effectively classify flight delays [9–15]. The stack autoencoder, comprising an encoder and decoder, learns data features and represents statistical characteristics of training samples. After training, the encoder model is saved, and the decoder is deleted, making it useful for feature extraction in new machine learning models [9,10,12,13]. The three autoencoder architectures (multilayer perceptron, logistic regression, and vanilla) handle data reconstruction differently after training, and the vanilla autoencoder outperforms the others in terms of accuracy, precision, and recall. This is due to factors like the use of the same input and output, the Adam optimizer, and the

mean squared error loss function. The impact of the Vanilla procedure on predicting flight delays is significant, with a 90% true-positive prediction for delayed flights and a 70% true-negative for not delayed flights. However, further modifications to the architecture's parameter settings could improve misclassified classes and positively impact air transportation systems in reducing false negative and false positive predictions in the flight monitoring system.

10.3 A Hybrid Voting Ensemble Approach for Flight Delay Classification

In this Chapter, we developed a hybrid ensemble Machine Learning approach for flight delay prediction, incorporating a feature sampling technique. The method generates a pool of candidate classifiers as weak learners, which are then combined through hard voting to form a strong committee. The aim is to assess the efficacy of this hybrid ensemble compared to other benchmark ensembles. Ensemble learning involves training multiple weak learners individually and combining their outputs for improved overall performance. The proposed approach addresses individual weaknesses of Machine Learning models by merging their final performances and employing majority voting for the final decision. We compared the hybrid ensemble (HVEM) with average, boosting, and stacking ensemble algorithms. HVEM outperformed other baseline techniques in precision, recall, accuracy and F1-score. The integration of sampling techniques and hard voting on weak heterogeneous learners in a binary classification task contributes to HVEM's higher performance. Additionally, the ensemble tackles bias and variance problems in the data, making it more robust in the classification task. HVEM provides valuable insight to stakeholders by reducing false positives and false negatives in flight delay predictions.

10.4 A Deep Bi-LSTM Approach for Flight Delay Modelling and Simulation

In this Chapter, we employed a min-max normalisation procedure to normalize the data and assess the performance of the Bi-LSTM for binary classification of flight delays using the US BTS dataset. Bi-LSTM, a powerful Deep Learning approach, efficiently analyses and extracts crucial data features for predictive tasks [31]. The proposed Bi-LSTM is fully connected and uses dropout as a regularization technique to prevent overfitting. The dataset is split into 70% for

training and 30% for testing to validate the model's performance with an unknown dataset portion. Thorough experiments were conducted using the BTS data, and the approach's performance was evaluated using various metrics, comparing it against a benchmark unidirectional LSTM model. The Bi-LSTM outperformed with an accuracy of 97% and an MCC of 99%, indicating that the model's predicted classes are 99% close to the original ground truth in training samples. The classification task displayed more than 90% true positive and negative predictions, with only a few misclassifications. This reliability enables effective resource allocation in airports and aids service providers in managing passenger flight delays. However, the model's performance was considered slow due to the forward and backward direction during training, a bottleneck that could be improved using high-performance computing facilities.

10.5 Social Ski Driver Conditional Auto-regressive Deep Learning for Flight Delay Prediction

In Chapter 8, a novel optimisation-aware Deep Learning model called Social Ski driver conditional autoregressive (SSDCA) Deep Learning was developed for flight delay prediction. This approach utilises Yeo-Johnson data transformation techniques in the pre-processing stage to stabilize the data and reduce variance. The main goal is to address the complexity of the relationship in the US BTS dataset using integrated optimisation and deep neural networks for pre-processing, training, and prediction of flight delays. The input data, containing time, numerical, and categorical attributes, undergoes missing value computation using average and frequency methods. Yeo-Johnson transformation is then applied to improve data symmetry, and a correlation table helps select highly correlated features for feature fusion. Deep Recurrent Neural Network (DRNN) is employed for feature fusion and target class determination due to its effectiveness with inputs of varying length. During training, the SSDCA optimizer is utilized to find optimal weights, accommodating the data's distributed resources and diverse characteristics. The SSDCA-based Deep Learning approach stands out by finding global best solutions through additional optimisation algorithms and multi-objective functions, enhancing the deep neural network classifier. This dynamic regulation of combined weights enhances the learning process and prediction results. Experimental results demonstrate the superiority of the

SSDCA-based Deep Learning model over other meta-heuristic and benchmark approaches, evidenced by lower MSE, RMSE, MAE, and higher accuracy values. The SSDCA optimisation contributes to improved model training by selecting optimal global weights and biases. The DRNN's ability to process inputs of any length and approximate any function adds to the model's effectiveness. SSDCA-based Deep Learning shows superior performance in predicting the total number of delayed flights for January and February 2020, providing valuable insights for airline process managers to allocate resources and plan peak periods. The proposed model outperforms other benchmarks, making it a reliable tool for flight delay prediction.

10.6 Prediction of Flight Delay using Deep Operator Network with Gradient-Mayfly Optimisation Algorithm

In Chapter 9, a novel approach is devised by extending the state-of-the-art DeepONet algorithm, initially designed for solving partial differential equations (PDEs), to the domain of flight delay prediction. DeepONet learns differential operators directly from data, enabling accurate predictions without explicitly solving the PDE. While DeepONet has shown promise in various fields, its applicability for flight delay prediction using time series datasets like the US BTS on-time performance data has been relatively unexplored. This study evaluates whether an optimized version of DeepONet, integrated with the Gradient Mayfly Optimisation algorithm, can enhance predictive modelling for flight delays. The proposed algorithm, named Political Social Ski Driver Conditional Autoregressive (PSSDCA) integrated with Deep Residual Network (DRR), uses DeepONet for feature fusion and optimizes the learning rate during training. Comparisons with other benchmark algorithms reveal that the optimized DeepONet outperforms traditional DeepONet and other approaches, achieving lower prediction errors with an MSE of 0.0058, RMSE of 0.0765, MAE of 0.0049 and MAPE of 0.0043. The optimised DeepONet utilises the Mayfly optimisation algorithm to select the best learning rate during training, which improves feature fusion and model training. This approach positively impacts flight delay prediction and monitoring, reducing errors that can lead to disruptions and increased costs. The proposed model fosters trust and loyalty from air passengers while aiding operational managers in making better decisions to mitigate delays' negative effects. Figure

9-4 and Figure 9-5 demonstrate the superior performance of the optimised DeepONet compared to existing approaches, indicating its potential for integration into flight delay and monitoring systems. As a generic strategy, the proposed approach can be applied to various domains with time series features to tackle critical issues, such as unnecessary financial losses and improving service provisions for better customer experiences. The findings offer a foundation for the development of effective, reliable, and high-performing predictive modelling systems in various industries. As a future direction, the proposed optimised DeepONet can be continuously improved by incorporating more data from different regions to understand variations in delay causes worldwide. Although validated with the US BTS data in this study, the model's applicability in global scenarios can be further explored. Ultimately, this research offers a valuable solution for flight delay prediction and addressing on-time datasets with time series features in diverse applications, leading to improved operational efficiency and customer satisfaction.

10.7 A Comparative Analysis of the Proposed Methods and Generalisation

During the period of this thesis, it has been discovered that the development of Machine Learning models for any predictive analysis task, like the flight delays prediction, requires critical analysis of the features and training the algorithms is better when optimisation algorithms are applied in deciding the learning rate instead of using a fixed learning rate. From the experiment conducted in this thesis. It is evidently clear that the performance and generalisation of a model in flight delay prediction are affected by not just the algorithm's robustness but the entire pipeline for the model development as seen in the traditional Machine Learning and the ensemble models. The results show the strength and weaknesses of traditional Machine Learning when the training samples are much as compared to the Novel Deep Learning models. However, the novel models' performances are also affected by the pre-processing and type of hyperparameter tuning adapted and applied. This is evident in the two-optimisation data-driven approach developed in this thesis. Despite all the excellent results achieved from the Deep Learning models for predicting flight

delay, as similar in previous studies, the datasets need to be further enriched by including other regional data to better compare the models across different instances. The data quantity from different scenarios has been identified as a common issue of Machine Learning tasks. To meet the industrial demand, there is a need to implement these models across real-world systems. As far as we can deduce, intelligent transportation system development still needs new technologists to motivate inventions.

In summary, each flight delay prediction algorithm proposed in this thesis has unique strengths and capabilities. The Deep Feedforward Neural Network offers a fundamental framework for capturing relationships in large datasets, while the Enhanced Stacked Autoencoder addresses the challenges of sequential data and high-dimensional features. The Hybrid Voting Ensemble combines diverse models for improved classification, and the Deep Bi-LSTM excels in modelling flight delays. The Social Ski Driver Conditional Auto-regressive Deep Learning approach leverages external factors, and the Prediction of Flight Delay using Deep Operator Network with a Gradient-Mayfly Optimisation Algorithm offers an innovative combination of a specialised deep learning architecture with advanced optimisation techniques. The choice of the most suitable algorithm will depend on the specific characteristics of the flight delay prediction task and the available data sources.

10.8 References

1. Balakrishna P., Ganesan R., Sherry L. Application of reinforcement learning algorithms for predicting taxi-out times. Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2009. 2009; : 255–261.
2. Balakrishna P., Ganesan R., Sherry L., Levy BS. Estimating taxi-out times with a reinforcement learning algorithm. AIAA/IEEE Digital Avionics Systems Conference - Proceedings. 2008; : 1–12. Available at: DOI:10.1109/DASC.2008.4702812
3. Sauvestre R., Duperier L., Leaf J. CS229 Final Report: Modelling Flight Delays. 2016; : 6. Available at: <http://cs229.stanford.edu/proj2016/report/DuperierSauvestreLeaf-ModellingFlightDelays-report.pdf>
4. Alla H., Moumoun L., Balouki Y. A Multilayer Perceptron Neural Network with Selective-Data Training for Flight Arrival Delay Prediction. Scientific Programming. 2021; 2021. Available at: DOI:10.1155/2021/5558918
5. McCarthy N., Karzand M., Lecue F. Amsterdam to Dublin Eventually Delayed? LSTM and Transfer Learning for Predicting Delays of Low Cost Airlines. Proceedings of the AAAI Conference on Artificial Intelligence. 2019; 33: 9541–9546. Available at: DOI:10.1609/aaai.v33i01.33019541
6. Natarajan V., Meenakshisundaram S., Balasubramanian G., Sinha S. A novel approach: Airline delay prediction using Machine Learning. Proceedings - 2018 International Conference on Computational Science and Computational Intelligence, CSCI 2018. IEEE; 2018. pp. 1081–1086. Available at: DOI:10.1109/CSCI46756.2018.00210
7. Guleria Y., Cai Q., Alam S., Li L. A Multi-Agent Approach for Reactionary Delay Prediction of Flights. IEEE Access. IEEE; 2019; 7(Cdm): 181565–181579. Available at: DOI:10.1109/ACCESS.2019.2957874
8. Moreira L., Dantas C., Oliveira L., Soares J., Ogasawara E. On Evaluating Data Preprocessing Methods for Machine Learning Models for Flight Delays. Proceedings of the International Joint Conference on Neural Networks. IEEE; 2018; 2018-July(C): 1–8. Available at: DOI:10.1109/IJCNN.2018.8489294
9. Bunrit S., Kerdprasop N., Kerdprasop K. Improving the representation of cnn based features by autoencoder for a task of construction material image classification. Journal of Advances in Information Technology. 2020; 11(4): 192–199. Available at: DOI:10.12720/jait.11.4.192-199
10. Chen M., Zeng W., Xu Z., Li J. Delay prediction based on deep stacked autoencoder networks. ACM International Conference Proceeding Series. 2018; Part F1482: 238–242. Available at: DOI:10.1145/3321619.3321669
11. Hu J., Zhang J., Zhang C., Wang J. A new deep neural network based on a stack of single-hidden-layer feedforward neural networks with randomly fixed hidden neurons. Neurocomputing. Elsevier; 2016; 171: 63–72. Available at: DOI:10.1016/j.neucom.2015.06.017
12. Bengio Y., Courville A., Vincent P. Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence. IEEE; 2013; 35(8): 1798–1828. Available at: DOI:10.1109/TPAMI.2013.50

13. Gulli A., Pal S. Deep Learning with Keras. 2017. 318 p. Available at: <https://www.packtpub.com/big-data-and-business-intelligence/deep-learning-keras>
14. Ravichandiran S. Hands-On Deep Learning.
15. Nguyen H., Kieu LM., Wen T., Cai C. Deep Learning methods in transportation domain: A review. *IET Intelligent Transport Systems*. 2018; 12(9): 998–1004. Available at: DOI:10.1049/iet-its.2018.0064
16. Mohareb F., Papadopoulou O., Panagou E., Nychas GJ., Bessant C. Ensemble-based support vector machine classifiers as an efficient tool for quality assessment of beef fillets from electronic nose data. *Analytical Methods*. Royal Society of Chemistry; 2016; 8(18): 3711–3721. Available at: DOI:10.1039/c6ay00147e
17. M. Al-Tabbakh S., M. Mohamed H., H. El Z. Machine Learning techniques for analysis of Egyptian flight delay. *International Journal of Data Mining & Knowledge Management Process*. 2018; 8(3): 01–14. Available at: DOI:10.5121/ijdkp.2018.8301
18. Khaksar H., Sheikholeslami A. Airline delay prediction by Machine Learning algorithms. *Scientia Iranica*. 2019; 26(5 A): 2689–2702. Available at: DOI:10.24200/sci.2017.20020
19. Dangut MD., Skaf Z., Jennions IK. Rare Failure Prediction Using an Integrated Auto-encoder and Bidirectional Gated Recurrent Unit Network. *IFAC-PapersOnLine*. Elsevier Ltd; 2020; 53(3): 276–282. Available at: DOI:10.1016/j.ifacol.2020.11.045
20. Kononova A V., De Wilde P., Coghill GM. Advances in computational intelligence (UKCI 2012). *Soft Computing*. 2013. 2181–2183 p. Available at: DOI:10.1007/s00500-013-1159-7
21. Yi J., Zhang H., Liu H., Zhong G., Li G. Flight Delay Classification Prediction Based on Stacking Algorithm. *Journal of Advanced Transportation*. 2021; 2021. Available at: DOI:10.1155/2021/4292778
22. Peng Y., Wu Z., Jiang J. A novel feature selection approach for biomedical data classification. *Journal of Biomedical Informatics*. Elsevier Inc.; 2010; 43(1): 15–23. Available at: DOI:10.1016/j.jbi.2009.07.008
23. Vonitsanos G., Panagiotakopoulos T., Kanavos A., Tsakalidis A. Forecasting Air Flight Delays and Enabling Smart Airport Services in Apache Spark. Springer International Publishing; 2021. 407–417 p. Available at: DOI:10.1007/978-3-030-79157-5
24. Antonio N., de Almeida A., Nunes L. Data in Hotel Revenue Management: Exploring Cancellation Drivers to Gain Insights Into Booking Cancellation Behavior. *Cornell Hospitality Quarterly*. 2019; 60(4): 298–319. Available at: DOI:10.1177/1938965519851466
25. Kiesil J., May IT. Predicting aircraft arrival times with Machine Learning. 2020;
26. Kadhum M., Manaseer S., Dalhoum ALA. Evaluation feature selection technique on classification by using evolutionary ELM wrapper method with features priorities. *Journal of Advances in Information Technology*. 2021; 12(1): 21–28. Available at: DOI:10.12720/jait.12.1.21-28
27. Liang H., Sun X., Sun Y., Gao Y. Text feature extraction based on Deep Learning: a review. *Eurasip Journal on Wireless Communications and*

- Networking. EURASIP Journal on Wireless Communications and Networking; 2017; 2017(1): 1–12. Available at: DOI:10.1186/s13638-017-0993-1
28. Rodríguez-Sanz Á., Fernández de Marcos A., Pérez-Castán JA., Comendador FG., Arnaldo Valdés R., París Loreiro Á. Queue behavioural patterns for passengers at airport terminals: A Machine Learning approach. *Journal of Air Transport Management*. 2021; 90(December 2019): 1–17. Available at: DOI:10.1016/j.jairtraman.2020.101940
 29. Maxson RW. Prediction of Airport Arrival Rates Using Data Mining Methods. 2018;
 30. Peng Y. A novel ensemble Machine Learning for robust microarray data classification. *Computers in Biology and Medicine*. 2006; 36(6): 553–573. Available at: DOI:10.1016/j.combiomed.2005.04.001
 31. Karim F., Majumdar S., Darabi H., Chen S. LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access*. 2017; 6: 1662–1669. Available at: DOI:10.1109/ACCESS.2017.2779939
 32. Lu L., Jin P., Pang G., Zhang Z., Karniadakis GE. Learning non-linear operators via DeepONet based on the universal approximation theorem of operators. 2021; 3(March): 31–33. Available at: DOI:10.1038/s42256-021-00302-5
 33. Sharma Priyadarshini M., Venturi S., Panesi M. Application of DeepOnet to model inelastic scattering probabilities in air mixtures. 2021; : 1–13. Available at: DOI:10.2514/6.2021-3144
 34. Wang S., Wang H., Perdikaris P. Improved architectures and training algorithms for deep operator networks. 2021; : 1–40. Available at: <http://arxiv.org/abs/2110.01654>
 35. Cuomo S., Di Cola VS., Giampaolo F., Rozza G., Raissi M., Piccialli F. Scientific Machine Learning Through Physics-Informed Neural Networks: Where we are and What's Next. *Journal of Scientific Computing*. 2022; 92(3). Available at: DOI:10.1007/s10915-022-01939-z
 36. Bisandu DB., Moulitsas I., Filippone S. Social ski driver conditional autoregressive-based Deep Learning classifier for flight delay prediction. *Neural Computing and Applications*. Springer London; 2022; 34(11): 8777–8802. Available at: DOI:10.1007/s00521-022-06898-y

Chapter 11: Conclusion and Future Work

This thesis proposes, develops, explores and evaluates new data-driven approaches for flight delay classifications and predictions. A United States Bureau of Transportation Statistics (BTS) dataset and records were employed to validate the algorithms proposed. The proposed approaches focused on finding and resolving some data irregularities in the BTS dataset, such as irregular patterns and trends, missing values, complex relationships among features and invalid data. It can be a major bottleneck for traditional classification and prediction tasks using Machine Learning algorithms. We have observed interesting results from the performance evaluation of the developed models. Having the best prediction model depends on the scope of the interrelationships among the features considering all the combinations of parameters and reasons for the delay. The model's performance evaluation was challenging because of the different parameterisations adapted to them and the variety of metrics used. However, the performance of the proposed approaches outperformed the existing ones seen in the literature. The overall research findings reveal that a robust and advanced flight delay monitoring and prediction method is viable for developing smart Artificial Intelligence data-driven flight delay models. The Optimisation driven DeepONet strategies, specifically the proposed mayfly optimisation-driven DeepONet, outperform other methods regarding minimum prediction error rates. The validation results suggest that the Mayfly optimisation-driven DeepONet model can predict flight delays with more than 90% reliability with an overall minimum error of 0.0043 in the US BTS dataset studied in this research. Compared to the other classical Machine Learning and meta-heuristic algorithms for classification and prediction of future estimated values, the mayfly optimisation-driven DeepONet algorithm performs better because of many characteristics, such as the optimised feature fusion techniques. Secondly, instead of training the model with a fixed learning rate, the algorithm utilised the mayfly greedy behaviour towards getting the best global best in the entire sample as the current learning in each iteration which means optimising the learning process of the whole prediction procedure.

11.1 How the Research Aim and Objectives Were Answered

As mentioned in Chapter 1, this research aims to develop a data-driven novel Deep Learning and Machine Learning techniques for flight delay classification and prediction for improved air passengers experience. The following objectives will assist in achieving the aim.

Objective 1: To conduct a systematic literature review on the application of data analytics to improve the air passenger experience through reliable flight delay awareness and investigate the existing methods and approaches for flight delay classification and prediction modelling. To find gaps in the existing studies by understanding the limitations and emphasising the countermeasures' design.

Evidence 1: In Chapter 2, a systematic review of the flight delay problem has been previously addressed, and the existing methods were studied to identify the gaps. Also, in all the technical chapters related work sections (Chapters 4-10). A review of flight delay classification and prediction techniques. All the literature reviews concentrate on the solutions' impact on improving the reliability of flight delay prediction. The open literature review reveals the need for more advanced flight delay modelling and analysis techniques. It would assist in dealing with the current and future datasets available from the airlines containing huge amounts of information that mere eyes or traditional statistical approaches cannot analyse. Chapters 2, 4, 5, 6, 7, 8 and 9 highlighted the existing Machine Learning methods' drawbacks in handling flight delay classification and prediction problems.

Objective 2: To perform data pre-processing to understand and quantify the complexities and distribution in the US BTS dataset.

Evidence 2: In Chapters 3, 4, 5, 6 and 7, we present an exploratory data analysis and pre-processing procedures to help understand the best fitting for each of the models to be developed, which explain and quantifies the various complexities and distribution intrinsic in the US BTS dataset. Employ the BTS dataset to develop a reliable and robust data-driven predictive model. It is important to interpret the data record, remove the noise attribute irrelevant to the set aim and perform feature extraction. Also, collecting the non-delay instances of the target variables is important for learning the model and evaluation. The appropriate

learning scenarios for the problem must be defined with a performance metric reflecting real-world demand. The challenge of predicting flight delays using real-world dataset samples is the presence of complex irregular trends and patterns, which affects temporal feature learning. The current methods for handling these data inconsistencies are understandable for a small sample size of data on simple machine-learning classification models. However, it is unclear if the existing approaches can improve the model performance when trained on huge time series datasets such as the BTS. Hence, carrying out exploratory data analysis (EDA) was considered important. The novel algorithms find the complex patterns and transform the BTS dataset for flight delay predictive modelling. This analysis assists in identifying the features that have a direct link to the flight delay, causing inconsistency in the information and processes of managing the passengers. It was also revealed that not all airlines and airports have the same impacts from all the factors causing the flight delay, and proper awareness of the passengers can improve their loyalty to the airlines.

Objective 3: To identify the different data features, relations and impacts on the flight delay.

Evidence 3: In Chapter 3, the result of the EDA shows the BTS dataset features and their relationships through the correlation matrix. The impact of the features on the target class is indicated by the correlation value, meaning the higher correlation value means a higher impact feature on the flight delay.

Objective 4: To design, implement, evaluate and validate different architectures of Deep Learning methods for flight delay classification and prediction.

Evidence 4: Objectives 2 and 4 work concurrently. Therefore, the proposed algorithms implemented for the flight delay predictive modelling, dealing with the complex relationships in the data for the classification and prediction problem, is essential. The US BTS datasets employed for training and testing the proposed algorithm are for different years. The data contains two major partitions the individual flight delay causal features forming the independent variables and the target class making the dependent variable. We used the sample instances of the ground truth dataset to test each model's ability to predict the future occurrences

of flight delays during validation. The dataset target comprises two major classes for either there is a delay or not-delay, and the modelling task is to either build a model to classify these delays with higher accuracy and reliability or a model that can accurately predict the total number of delayed flights for each in a particular month with minimum error. After transforming the dataset in objectives 2 and 3, the data is then split into 70% for training and 30% for model testing. In other cases, the data is partitioned into 60:40, 70:30, 80:20 and 90:10 for the train/test ratio. It is for better exploitation and exploration of the model. Chapters 4, 5, 6, 7, 8 and 9 present an approach for flight delay classification and prediction to improve air passenger experiences and manage unwanted financial losses.

The validation of the developed models was carried out using the data ground truth. It ensures that the quality of the proposed flight delay predictive model is confirmed. The ground truth is the actual record of the flight on-time dataset that defines when there is a delay in the data. This definition is determined by the International Air Transport Association (IATA) and is globally recognised as the benchmark condition for indicating the presence of flight delay. The results of the models were validated against the definition contained in the ground truth record (sample instances leading to delay of flights). Analysing further reveals the performance of the proposed model in predicting the total number of flights delayed every day in a month. Predicting the flight in advance within a reasonable period, not too early or too far from the decision-making period, to allow enough time for proper mitigation strategies by the stakeholders (e.g., 2-3 days before the flight day, until the flight have taken place).

Objective 5: To recommend the most appropriate Deep Learning architecture implementation and approach for flight delay prediction based on our experiment.

Evidence 5: In Chapters 4, 5, 6, 7, 8 and 9, we have recommended the most performing models based on the performance showcase in the given task of flight delay classification and prediction. This recommendation follows the extensive experiment conducted in this research. Further recommendations on further improving the performances of the current models proposed in this research are stated accordingly in the thesis.

11.2 Contribution to Knowledge

During this research, significant contributions to knowledge have been made. This research, to this end, has made the following contributions to knowledge.

a). In this research, we reviewed literature systematically that addresses the problem of air passenger experience through flight delay from the perspective of data scientists across the academia and industrial sectors to understand current trends from the existing literature:

The review of the relevant literature contributes to knowledge by providing an understanding and familiarisation with the state-of-the-art research trends in Machine Learning and data analytics related to flight delay predictive modelling. It also confirms the increase in the complexity and irregularities in the shape and records of the BTS data, which means high demand for analytical methods to address the challenges posed even in the flight delay classification and prediction tasks leading to poor performance of the predictive modelling algorithms.

b). Exploratory Data Analysis (EDA) from the perspective of flight on-time dataset: The analysis of the US BTS dataset through the visualisation and graphs plots to discover the patterns, hidden information, outliers and missing values. The analysis improved the overall knowledge of the variables through the statistics, such as finding means and medians. The graphical display of the flights across different airports and airlines shows the average of the delay and not-delayed class to provide the required insight on the BTS datasets. This analysis information is important for future direction and comparative studies of the datasets for different years and regions.

c). Design and Implement an Approach for flight delay classifications and predictions using Deep Feedforward Neural Network: Developing a predictive model for classifying flight delays using the US BTS dataset is one of the contributions of this research that remains significant. The dataset utilised contains many outliers and inconsistent information that can affect the model's overall performance. A pre-processing technique was applied to remove the missing values and outliers. Then a well-known feature extraction method called the filter-based feature extraction approach was applied to select the most influential feature on the target class. Then a deep neural network was

successfully utilised for the individual flight delay classification and prediction. This algorithm can also classify any on-time performance datasets, such as train or bus datasets.

d). Designed and developed an enhanced deep stack-autoencoder for flight delay prediction: Another contribution worth mentioning is developing a predictive modelling technique based on deep auto-encoder by comparing three different modified architectures known as multilayer perceptron autoencoder, logistic regression and vanilla based on their predictive performance of the flight delay prediction. Autoencoder parameters were used to execute network training to understand better the link between space and time and information gained from the BTS data. The vanilla autoencoder's performance reconstructs and classifies the delay with higher precision, recall, accuracy and F1-score using the different number of hidden layers, input dimensions and epochs. The vanilla autoencoder results on the classification of the flight delays indicate it is relevant as an algorithm capable of learning varying shapes of input and improving performance when fine-tuned appropriately with the correct hyper parameterisation.

e). Developed a hybrid ensemble Machine Learning algorithm for flight delay modelling and implemented a Bi-LSTM model for flight delay classification: Another significant contribution is the development of a hybrid ensemble learner using hard voting techniques. The algorithm focuses on improving the classification accuracy of the predictive model using the US BTS dataset. In addition, the proposed hybrid data-driven flight delay predictive model utilises resampling techniques and combines the base classifiers to form the classification committee based on a high probability of classifiers. This approach solves the problem of a lazy learners algorithm by combining their predictive power to form a strong committee of ensembles. The model's performance is robust, with over 90% predictive accuracy. The Bi-LSTM methods developed in this research performs well based on the evaluation metrics used and have a higher Mathews correlation coefficient, which means there are few differences between the actual or ground truth and predictive models.

f). Proposed and developed a new optimisation-aware data-driven approach for flight delay prediction: The Deep Learning approach is designed to capture patterns in the BTS dataset and perform transformation using Yeo-Johnson. A deep recurrent neural network is utilised for training the feature fusion, and then the social ski conditional value at risk optimisation assists in selecting the best weight and biases during the model's training. The model is used to predict total flight delays for one month. The results from the model validation on the BTS dataset were promising and thus contributed significantly because the optimisation algorithm's efficacy is revealed based on the prediction's minimum error.

e). Developed a novel flight delay prediction approach using the DeepONet and mayfly optimisation algorithms: This newly developed data-driven optimisation aware approach extends a state-of-the-art model for solving approximation problems with partial differential equations. The model involves a deep residual network integrated with political social ski driver conditional autoregression for feature fusion for obtaining optimum weight and gradient mayfly optimisation algorithm for training and predicting the flight delay. The new approach is validated using a real-world dataset. Exploring the BTS dataset for developing a robust optimisation-aware data-driven DeepONet is a significant contribution because of its generalisation ability and the possibility to achieve high-performance results in other related predictive modelling problems.

11.3 Intellectual Contribution and its Impact

In this research, we proposed novel algorithms and methods for flight delay classification and prediction based on data-driven optimisation aware Deep Learning (DL), Machine Learning (ML), Ensemble learning (EL) and Stack autoencoder (SA). It presents alternative approaches to solving flight delay classification and prediction problems using the real-world dataset from the United States on-time performance data record lacking in the existing literature.

The research focuses on developing a flight delay predictive model with a unique and robust ability to provide information about the flight before the scheduled

departure or arrival time. The model can be integrated into flight delay decision systems when fully validated in more real-world situations. Therefore, improve passenger experience and reduce unnecessary loss of money by the airlines, proper resource allocations and mitigation procedures to curtail processes disruption, which benefits the airline's stakeholders.

Furthermore, the research impacts both academia and the airline industries. The impacts in academia can be disseminating the research output as published papers in conferences and journals. The industrial impact is contributed towards effective flight delay predictive modelling. Additionally, the research not only proposes flight delay prediction solutions for the air transportation system by the gaps identified, focusing on improving airline and airport passenger experience alone. It can be adapted and implemented in other domains faced with the problem of effective predictive modelling problems. Therefore, the research outcome can generally be employed in industries to develop robust classification and predictive modelling solutions for similar problems.

The publications are available at '<https://orcid.org/0000-0002-5723-8429>.' Cranfield's Intellectual Property (IP) Policy governs the ownership of rights arising from research and/or materials produced by Cranfield's members of staff and students, supports funding applications to public and private sector partners, and establishes objectively fair and equitable criteria for the exploitation of knowledge. Cranfield aims to provide support services through the Research & Innovation Office (RIO) to promote the creation of Intellectual Property (IP) whilst seeking to maximise the impact and commercial exploitation of the resulting.

11.4 Limitation of the Research

One notable limitation of this research is the exclusive use of the US BTS dataset for training, testing, and validating the developed models. Consequently, the findings and predictions are limited to a perspective focused solely on domestic flights within the United States. To enhance the comprehensiveness and generalisability of the study, future research could consider incorporating data from other regions or international flights to provide a more holistic view of flight delay prediction.

11.5 Future Work

During this research, several areas have been identified for further future research. They are as follows:

- a). This research can be extended by including datasets from different countries and regions to help understand the different regional impacts and patterns of flight delays.
- b). In the future, weather-related factors can be considered in the future to assist in predicting weather-impacted flight delays. Though our dataset considered weather delays with extreme event problems, this would be a good gap in the future developed models.
- c). In the future, we hope to develop a model that would predict the total number of delays for a month, the delay for a short term, and specific time frames.
- d). Further studies can be done to improve the generalisation of the robust models developed in this research. The Physics-Inform-Machine-Learning (PiML) and the use of explainable and interpretable Machine Learning or Deep Learning can improve the general community's understanding of the prediction outcome.
- e). All the Deep Learning models developed in this research have the potential to be improved through additional variables and hyper parametrisation of the models' architectures. Also, adding other environmental factors can improve the model's performance.
- f). The concept of transfer learning has been recently considered a promising area because it provides room for knowledge learned from a different scenario of Machine Learning applied in other domains, thereby improving inter-domain knowledge sharing.
- g) Grid search can be a valuable tool in automating the feature selection process and determining the optimal combination of features for improved model development. By systematically exploring various feature selection techniques and their parameter combinations, grid search helps identify the most effective features contributing to the model's predictive performance.