Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 59 (2017) 257 – 262

www.elsevier.com/locate/procedia

The 5th International Conference on Through-life Engineering Services (TESConf 2016)

Analysis of short form maintenance records for NFF using NLP, phrase matching, and Bayesian learning

Jonathan G. Pelham[a,*], Chris Hockley[a]

[a]*Through-Life Engineering Services Centre, Cranfield University, College Road, Cranfield, MK43 0AL, UK*

* Corresponding author. Tel.: +44 (0) 1234 750111 x5756 ; *E-mail address:* j.g.pelham@cranfield.ac.uk

Abstract

No Fault Found (NFF) is a well discussed phenomenon within the maintenance sector but which requires work to quantify how much of an issue it may be and provide metrics by which it may be tracked and various approaches to its reduction evaluated. Previous studies have relied on expert classification to identify NFF, however this approach is time consuming and costly. Maintainer classification (MC), expert classification (RC), phrase matching (PM), and Bayesian matching (NBPM) are all evaluated and contrasted as methods to identify NFF. The results demonstrate the utility of all 4 methods and discusses their place within a maintenance ecosystem.

© 2016 Elsevier B.V. This is an open access article under the CC BY license
(http://creativecommons.org/licenses/by/4.0/).

Peer-review under responsibility of the scientific committee of the The 5th International Conference on Through-life Engineering Services (TESConf 2016)

Keywords: NFF ; No Fault Found ; NLP ; Maintenance ; Bayes

1. Introduction

The historical context of NFF has moved from one of anecdote to a well discussed part of maintenance. The NFF cycle can be seen in Figure 1 which depicts the process from an initial fault report, through a work order being raised, to diagnosis of no fault found, to return to service. This process can continue around the cycle at different levels of maintenance causing a waste of personnel and materials resources and uncertainty regarding the true health state of the asset. Many authors acknowledge the problem exists to some extent but quantifying the level of NFF and creating a metric has proved difficult. The large array of potential causes and differing opinions of the true reasons for NFF can make it difficult to create a true picture of the level of NFF within maintenance operations[1–3]. A MSc thesis by Stewart Roke manually classified MWOs(Maintenance Work Orders) for NFF and suggested a true level of between 8% and 16% within the maintenance of the RAF Harrier fleet during the period January 2005 to December 2008(p82)[4]. Roke was an expert technician who had worked on Harrier for a great many years and so could apply expert judgement to the work carried out recorded within the MWOs. This classification work when combined with the maintainer NFF classification at the time of the MWO into the LITS system provides a picture of how the maintainer and an expert of the fact view the NFF content of the MWO. This presents an interesting opportunity to evaluate different methods of automatic classification as the coding done by Roke can be used as a comparison with various algorithms in order to provide credibility. As algorithms improve they can be used with other fleets to gain a picture of the level of NFF more widely and to evaluate the effectiveness of different methods intended to reduce the NFF rate within maintenance.

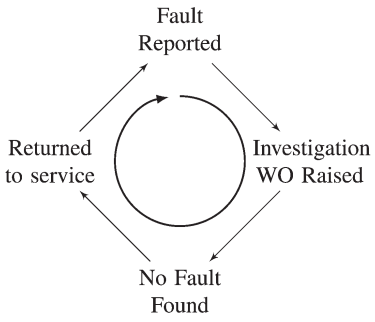


Fig. 1. No Fault Found cycle

2212-8271 © 2016 Elsevier B.V. This is an open access article under the CC BY license
(http://creativecommons.org/licenses/by/4.0/).
Peer-review under responsibility of the scientific committee of the The 5th International Conference on Through-life Engineering Services (TESConf 2016)
doi:10.1016/j.procir.2016.10.123

1.1. Previous work

> Text Mining is the discovery by computer of new, previously unknown information, by automatically extracting information from different written resources. (Marti Hearst [5])

The use of programming to create methods to classify, tag, and summarise otherwise unstructured text is the field of natural language processing and there are several examples of prior efforts to apply NLP to maintenance records. McKenzie et al[6] and Bokinsky et al[7] both made use of the Python NLTK(Natural Language Toolkit) to try to extract meaning from the maintenance records of aircraft. They both demonstrated the difficulty of working with part numbers which within a single aircraft type may have some order but will still evolve with the aircraft to make the recognition of text elements as part numbers a difficult process. This difficulty does not apply to the current work as at this stage it was decided not to attempt to identify part numbers within the text field. Butters et al[8–10] investigated the use of Named Entity Recognition to extract meaning from maintenance records. Their approach was evaluated for this NFF work but due to the short length of the MWO field (less than 250 characters) their method did not yield significant advantages. Future work is expect to investigate the application of their TR tool to longer maintenance records where it will be better suited. Various machine learning approaches have been tried to extract information from text with some success[11,12]. The python NLTK features a naive Bayes reasoner that has been used several times as part of analysis to estimate member class of some text sample. This approach has been used to estimate the impact of media statements on the equity of a particular firm[13] and to evaluate sentiment within statements[14].

2. Method

2.1. Problem

This study is interested in ways to classify maintenance work orders into categories relevant to the study of NFF. These work orders typically have useful information regarding the asset being examined, the reported issue (and expected fault) and the maintainers report on work done. In the data under study the organisation has instituted a practice of voluntary coding by maintainers to classify work orders as NFF or Fault Found. Our interest was to take these MWOs and create methods by which the historic levels of NFF could be assessed and the potential effectiveness of the voluntary NFF coding could be evaluated. This project has several potential approaches which fall into two main categories of Manual and Automated classification. Manual classification as discussed by Beniaminy et al [15] and others is a difficult undertaking and requires intimate expert knowledge of the maintenance asset which is the focus of the study. In the long term the development of NFF experts and review of maintenance performance with a view to better understanding the causes of NFF may be the best solution however to reach this point an automated method may be better suited to identifying potential NFF incidents which can then have resource dedicated to them. Manual classification of the thousands of maintenance records each year would likely be a waste of resource. We will compare three methods of classifying maintenance work orders for NFF. The expert classification by Roke was based on his interpretation of the work done notes written by the maintainer. The phrase matching method is an attempt to replicate this method using automated methods. In order to evaluate the performance of the automated method compared to previous work the dataset used by Roke in his MSc thesis was obtained to form a basis of comparison. This enabled a three way comparison between Roke's expert classification, the classification of MWO as NFF at source by RAF maintainers and the proposed phrase matching method.

2.2. Maintainer Coding

The maintainer coding method relies on expert maintainers correctly coding their tasks on completion NFF if they believe a NFF has occurred. Guidance is given to the maintainer on coding during their training and is also provided as part of the documentation for the use of the maintenance logging system[16].

2.3. Expert Classification

Roke used expert knowledge to classify short form maintenance records as NFF or FF within the RAF Harrier fleet 2005-2008. This process required a large amount of manpower effort and would likely be difficult to replicate without similar expert knowledge for the maintained asset under investigation. Common phrases denoting NFF were identified and multiple instances found of possible failure of the maintainers to code instances correctly as NFF. The phrases identified in Rokes study are shown in table 1.

Table 1. Alternative descriptors used instead of NFF within maintenance records. Taken from Study by Roke[4] (Table 9 P80)

Serial	Description
1	No Fault Found
2	Fault cleared during Investigation
3	Unable to reproduce reported fault
4	Fault not found
5	NFF or nff
6	No apparent damage
7	No faults apparent
8	No fault indications
9	No failure codes on ...
10	Unable to replicate fault
11	Unable to reproduce original fault symptom
12	Cannot reproduce fault
13	No problems reported

2.4. Phrase Matching

In order to construct an algorithm to find potential NFF incidents within the maintenance data extensive use was made of the Python programming language, the natural language tool kit, and the open source Python data analysis library Pandas. The specific versions used can be found in Appendix A. The maintenance data received is passed into a pandas dataframe

using some custom Python code so it can be analysed. The data then undergoes a number of steps.

1. Pre Processing
   (a) The data is reduced to a subset based on the coordinated date of the work order.
   (b) A custom created dictionary is applied to correct spelling mistakes. The nature of manual text entry into a terminal within a maintenance bay while in a rush makes this step important for classification accuracy.
2. Classification
   (a) Natural Language Tool Kit splits the text entries into word tokens
   (b) The phrase matching algorithm is applied using the specified phrases to classify the work orders as NFF or FF by matching phrases found from the word token tuples. e.g. words, bigrams, and trigrams
   (c) See Figure 2 for details of the processing performed on each maintenance work order.
3. The data is split into domains. For this study the Forward(FWD) Domain from Rokes work was examined.

A full list of the the phrases used by the algorithm to classify MWOs as NFF can be found in Appendix B. The algorithm makes use of the Natural language tool kit to split the maintainer entered text into words and phrases which can then be compared to the phrases selected as signifying NFF. The use of phrases as a matching technique is dependant on the presence within the maintenance culture of a shared report structuring system. Where maintainers use the same phrase it is a requirement for the algorithm that they mean the same thing. In practice this may not always be the case and there will always be some ambiguity. The more fragmented the linguistic culture of the maintenance organisation, the more difficult it will be to identify phrases which signify NFF from the maintenance data. The use of interviews to attempt to identify local variation in the description of NFF problems has bee suggested as a method to capture this local variation and it will likely be effective. These discussions of NFF descriptions may cause maintainers to change their use of language in unexpected ways but it is likely still a positive step as it will identify some phrases and also get the maintainers thinking about NFF and their intention when they are writing their maintenance report.

2.5. Naive Bayes Phrase Matching

The Naive Bayes algorithm[17] is based on the idea that the probability of an event can be determined based on the probability of it's constituent elements and that the probabilities of these elements are independent of each others state. This is why it is known as naive Bayes as where one element is true it is not unusual to find other elements true and so the state of elements can be dependant on the state of other elements. If one is True then the probability of the other elements being True is affected in a dependant relationship. Despite this problem Naive Bayes has been successfully used in natural language processing in several applications and in particular is associated with efforts to try to enable a computer to classify the opinion of a reviewer from the text of their article[18].

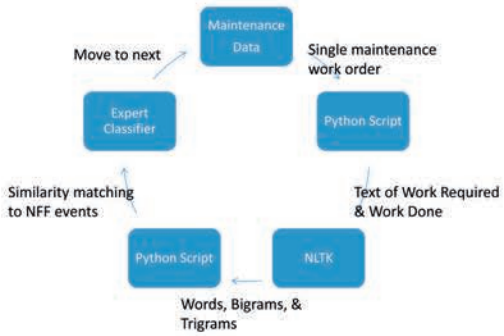


Fig. 2. Maintenance data NLP analysis process

Feature Function.
Several different feature functions were evaluated following guidelines within the NLTK book[19] and while all showed high classifier ability the bigram feature function showed the best performance by a narrow margin. Evaluation of the trigram and whole sentence feature functions suggested they may be susceptible to overmatching. Study in the area of machine learning using Artificial Immune Systems[20] has suggested that it is more appropriate to encourage feature selection so that there will be multiple supporting medium weight matches within a sample rather than one high weight match. The bigram feature function was therefore selected to evaluate.

Training and Evaluation.
The different feature functions were trained and evaluated using a randomly selected subset of less than 9% of the total dataset. They were separated into training, development, and testing sets in the ratio (62.5% : 25% : 12.5%). Each classifier was train several times with random data used each time to evaluate performance stability of the classifier. The bigram feature function and the built in NLTK naive Bayes classifier averaged an average accuracy of 93.96% with a standard deviation of 0.84%.

3. Results

The Roke classification and the maintainer classification were joined using their unique identifiers to produce a dataset for which every entry was classified by both methods. This was turned into a python pandas dataframe for ease of manipulation. Any MWO which were featured in one set but not the other were discarded. The rule based phrase matching method and the naive Bayes phrase matching algorithm were then both applied and logical indexing used to evaluate the correspondence between the methods and the MWO for which there was disagreement. These were then created into the two separate Venn diagrams shown in Figure 3 and 4. This was done to evaluate the overall match and overlap. The naive bayes and the rule based phrase matching were also compared to each other and had a high degree of agreement but this data is not included here for lack of space. The proportion of the different sets was calculated to allow the proportions to be understood without disclosing the total numbers of MWO. The classifiers developed were both applied to the full dataset in order to create a

full scale comparison. Venn diagrams were then created to examine the intersection of the different sets. These are shown in Figures 3 and 4 and are described in Tables 2 and 3. The results from the phrase matching algorithm are presented here in Figure 3 and Table 2. They demonstrate reasonable agreement with the results of Roke's classification and with the maintainer classification of the MWO. The results from the naive Bayes phrase matching algorithm are presented here in Figure 4 and Table 3. The naive Bayes phrase matching algorithm demonstrates an improved agreement with the results of Roke's classification and with the maintainer classification of the MWO over the performance of the rule based phrase matching algorithm. An assessment of the possible level of NFF from these four methods gives the following figures. The level of NFF in the dataset is between 7% and 20.89% depending on the classification method used. Maintainer Coding 7% , Roke Study 15% , Phrase matching method 20.35%, and Bayesian Phrase matching method 20.89% by work order.

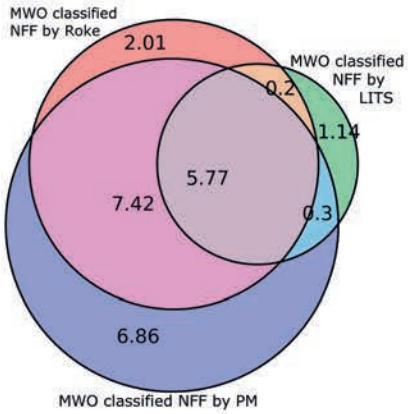


Fig. 3. Venn Diagram showing the Intersection of the sets of MWO classified as NFF by Roke, Maintainer, and the phrase matching method. To be read in coordination with Table 2 which gives the identity of each subset of the Venn diagram.

Table 2. Figure 3 set details. The different sets shown in Figure 3 are listed with their size as a proportion of the full maintenance dataset analysed. Uses rule based phrase matching

Subset	% of Set	Running Total %
RC (Roke Classification)	2.01	2.01
MC (Maintainer Coding)	1.14	3.15
PM (Phrase Matching)	6.86	10.01
RC & MC only. No PM	0.20	10.21
RC & PM only. No MC	7.42	17.63
MC & PM only. No RC	0.3	17.93
RC, MC & PM	5.77	23.7
RC total set 15.40% = 2.01 + 0.2 + 7.42 + 5.77		
MC total set 7.41% = 1.14 + 0.2 + 0.3 + 5.77		
PM total set 20.35% = 6.86 + 7.42 + 0.3 + 5.77		

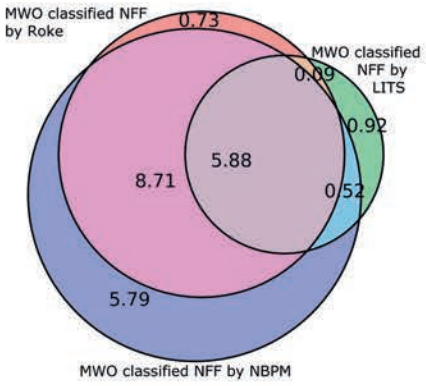


Fig. 4. Venn Diagram showing the Intersection of the sets of MWO classified as NFF by Roke, Maintainer, and the Naive Bayes phrase matching method. To be read in coordination with Table 3 which gives the identity of each subset of the Venn diagram.

Table 3. Figure 4 set details. The different sets shown in Figure 4 are listed with their size as a proportion of the full maintenance dataset analysed. Uses Naive Bayes phrase matching.

Subset	% of Set	Running Total %
RC (Roke Classification)	0.73	0.73
MC (Maintainer Coding)	0.92	1.65
NBPM (Naive Bayes Phrase Matching)	5.79	7.44
RC & MC only. No NBPM	0.094	7.53
RC & NBPM only. No MC	8.71	16.24
MC & NBPM only. No RC	0.52	16.76
RC, MC & NBPM	5.88	22.63
RC total set 15.40% = 0.73 + 0.094 + 8.71 + 5.88		
MC total set 7.41% = 0.92 + 0.094 + 0.52 + 5.88		
NBPM total set 20.89% = 5.79 + 8.71 + 0.52 + 5.88		

Table 4. Confusion Table of the results of the naive Bayes phrase matching algorithm

		Roke Classification		
		Positive	**Negative**	**Total**
NBPM Output	**Positive**	14.6%	6.31%	20.89%
	Negative	0.82%	78.29%	79.11%
	Total	15.4%	84.6%	100%
Sensitivity = 94.68%				
Specificity = 92.54%				

4. Discussion

Examining the intersection of the suspected NFF sets from the different methods depicted in Figures 3 and 4 and in Tables 2 and 3 shows the communality and difference between the different approaches. The true NFF rate may be somewhere between the results obtained by these methods but would be difficult to ascertain. It is interesting to note that the phrase matching method and the naive Bayes method come to similar con-

clusions regarding the level of NFF but do not exactly match. The phrases used in the phrase matching algorithm came directly from Roke's work and the 35% in predicted NFF level of the phrase matching method vs the Roke level suggests there is additional nuance within the text used to determine NFF status which has not yet been identified. The same applies to the naive Bayes method which suggests a NFF level 39% higher than than suggested by Roke. The higher level suggested by the Bayes classifier is especially interesting because unlike the rule based phrase matcher it was trained directly from a portion of the data classified by Roke. Thus the higher level either points to Roke using information from portions of the MWO not covered by this algorithm into his classification or to potential misclassification. Future study will focus on the MWO where the classifiers disagree with Roke in an effort to better understand the traits that make MWO more and less likely to be NFF. The performance of the classifier is good comparatively with a sensitivity and specificity both over 90% but future work will hopefully give insight into the way language is used and how to future improve classifier performance while gaining better understanding of how maintainers describe NFF issues.

Acknowledgements

We would like to thank the MOD for their kindness in enabling this research, Mr S Roke for allowing us to use his thesis data, and the EPSRC for funding the project with grant EP/I033246/1

Appendix A. Resources

1. Python programming language v2.7.9 https://www.python.org/
2. Natural Language Tool Kit v3.0.2 http://www.nltk.org/ (See the NLTK book for further details[19])
3. Pandas (Python data analysis library) v0.15.2 http://pandas.pydata.org

Appendix B. NFF Phrases

- 'no fault found', 'no faults found', 'no fault', & 'no faults'
- 'no fault code', '(code 01)' & 'fault cleared during investigation'
- '(code 49)', 'unable to reproduce reported fault'
- 'unable to reproduce', 'fault not found'
- 'nff', 'n f f', 'n.f.f .', & 'n . f . f'
- 'no apparent damage' & 'no faults apparent'
- 'no fault indications' & 'no failure code'
- 'unable to replicate' & 'unable to replicate fault'
- 'unable to reproduce original fault indication'
- 'cannot reproduce fault' & 'no problems reported'
- 'fault cleared during test'
- 'fault cleared itself during investigation'
- 'fault cleared during investigation'
- 'fault cleared during' & 'fault cleared itself'
- 'assessed serviceable' & 'confirmed serviceable'
- 'unable to identify fault', 'fault within limits' & 'reset and test ok'
- '13 - last item replaced - u/s item is r2'
- 'reseated ok', 'reseated satis', 'bit re-run satis' & 'system function ok'
- 'system function satis' & 'no fault after check'
- 'no fault after test' , "sst 's c/o satis" & 'sst c/o satis'

References

[1] Khan, S., Phillips, P., Hockley, C., Jennions, I.. Towards standardisation of no fault found taxonomy 2012;:–.
[2] Khan, S., Phillips, P., Hockley, C., Jennions, I.. No fault found events in maintenance engineering part 2: Root causes, technical developments and future research. ReliabEngSystSaf 2014;123:196–208.
[3] Khan, S., Phillips, P., Jennions, I., Hockley, C.. No fault found events in maintenance engineering part 1: Current trends, implications and organizational practices. ReliabEngSystSaf 2014;123:183–195.
[4] Roke, S.. Harrier no fault found reduction. Master's thesis; Faculty of Engineering and Computing, Coventry University; 2009.
[5] Hearst, M.. What is text mining. SIMS, UC Berkeley 2003;URL: http://www.jaist.ac.jp/ bao/MOT-Ishikawa/FurtherReadingNo1.pdf.
[6] McKenzie, A., Matthews, M., Goodman, N., Bayoumi, A.. Information extraction from helicopter maintenance records as a springboard for the future of maintenance text analysis. In: Trends in Applied Intelligent Systems. Springer; 2010, p. 590–600. URL: http://link.springer.com/chapter/10.1007/978-3-642-13022-9_59.
[7] Bokinsky, H., McKenzie, A., Bayoumi, A., McCaslin, R., Patterson, A., Matthews, M., et al. Application of natural language processing techniques to marine v-22 maintenance data for populating a cbm-oriented database. In: AHS Airworthiness, CBM, and HUMS Specialists' Meeting. 2013,URL: http://cbm.me.sc.edu/conference_pubs.html.
[8] Butters, J.D.. Terminology recognition in the aerospace domain. Ph.D. thesis; University of Sheffield; 2012.
[9] Butters, J., Ciravegna, F.. Authoring technical documents for effective retrieval. In: Knowledge Engineering and Management by the Masses. Springer; 2010, p. 287–300.
[10] Butters, J., Ciravegna, F.. Using similarity metrics for terminology recognition. In: LREC. Citeseer; 2008, p. –.
[11] Ciravegna, F.. Learning to tag for information extraction from text. In: Workshop Machine Learning for Information Extraction, European Conference on Artifical Intelligence ECCAI. European Conference on Artifical Intelligence ECCAI; 2000, p. 416–420.
[12] Ireson, N., Ciravegna, F., Califf, M.E., Freitag, D., Kushmerick, N., Lavelli, A.. Evaluating machine learning for information extraction. In: Proceedings of the 22nd international conference on Machine learning. ACM; 2005, p. 345–352.
[13] Yu, Y., Duan, W., Cao, Q.. The impact of social and conventional media on firm equity value: A sentiment analysis approach. Decision Support Systems 2013;55(4):919–926.
[14] Cohen, M., Damiani, P., Durandeu, S., Navas, R., Merlino, H., Fernández, E.. Sentiment analysis in microblogging: a practical implementation. In: XVII Congreso Argentino de Ciencias de la Computación. 2011,.
[15] Beniaminy Israel.; Joseph, D.. Reducing the "no fault found" problem: Contributions from expert-system methods. In: Aerospace Conference Proceedings; vol. 6. IEEE; 2002, p. 2971–2973.

[16] Ministry of Defence, . Dap 300a-01 section 1 leaflet 003 maintenance and fault recording, reporting, and analysis. 2013.
[17] Lewis, D.D.. Naive (bayes) at forty: The independence assumption in information retrieval. In: European conference on machine learning. Springer; 1998, p. 4–15.
[18] Dinu, L.P., Iuga, I.. The naive bayes classifier in opinion mining: in search of the best feature set. In: International Conference on Intelligent Text Processing and Computational Linguistics. Springer; 2012, p. 556–567.
[19] Bird, S., Klein, E., Loper, E.. Natural language processing with Python. " O'Reilly Media, Inc."; 2009. URL: http://www.nltk.org/book/.
[20] Cohen, I.R.. Tending Adam's Garden: evolving the cognitive immune self. Academic Press; 2000.



https://dspace.org/

Analysis of short form maintenance records for NFF using NLP, phrase matching, and Bayesian learning

Pelham, Jonathan G.; Hockley, Chris

2017-03-02
Elsevier
https://doi.org/10.1016/j.procir.2016.10.123
Pelham J, Hockley C, Analysis of short form maintenance records for NFF using NLP, phrase matching, and Bayesian learning, Procedia CIRP, Volume 59, 2017, Pages 257-262

Downloaded from a DSpace repository. https://dspace.org/