VIENNA 2018 TRA
A digital era for transport
solutions for society, economy and environment

Proceedings of 7th Transport Research Arena TRA 2018, April 16-19, 2018, Vienna, Austria

Opportunities for resilient rail system development using natural language processing.

Ngo, D b, Parkinson, H a *, Bamford, G a. Bamford J a, Rayson, P b.

[a]*Digital Rail Limited, Infolab21, Lancaster LA1 4WA, England*
[b]*Lancaster University, Bailrigg, Lancaster LA1 4YW, England*

Abstract

In this paper we examine a natural language and machine learning approach to help assess the quality of railway hazard logs. The focus is on highlighting red flags in the hazard log content to help improve the accuracy and quality of the contents and so the speed of risk reviews. Data is presented that indicate the approach has potential for significant savings in time and increased quality. The tool is one of a number that we are developing as part of an initiative to improve rail system development and operation by employing artificial intelligence (AI) to augment existing methods in the context of a wider system engineering approach. This will in turn lead to rail systems becoming more sustainable and resilient.

Keywords: Natural Language Processing; NLP: Machine Learning; Railway Safety; System Engineering, Artificial Intelligence

* Parkinson, H.. Tel.: +44 (0)7803 581 849.
E-mail address: hjparkinson@digitalrail.co.uk

1. Introduction

Many projects end up in trouble because they have not followed recognised systems engineering processes, as described by Elliot [2014] and this is indeed the main reason for having system engineering; it provides management with timely information regarding the health of a project. Although projects are normally set up to follow these processes, during delivery, drift in the day-to-day activities tends to result in a gradual migration to a state of non-compliance. The wider objectives of this research is to help identify any drift to non-compliance by applying Natural Language Processing (NLP) and Artificial Intelligence (AI) to the systems engineering lifecycle to monitor adherence to standards and best practice, in this case a railway safety hazard log. The idea behind the project is in essence to have intelligent actors monitoring system development process which will enable early insight into how well the project is on course in terms of requirements, safety, gate reviews and validation for example.

1.1. System engineering and requirements engineering

The use of AI in systems and software development has not been widely explored; however, the consensus is that it provides the best opportunity particularly in managing and augmenting the system development processes, Sommerville (1994).
Problems usually arise in complex system development when the amount of textual data that the company must process becomes too large. The volume and variety of the data increases even further as the need for collaboration grows, and with it comes the difficulty of managing the content, keeping up the text readability, the standard format and model consistency.

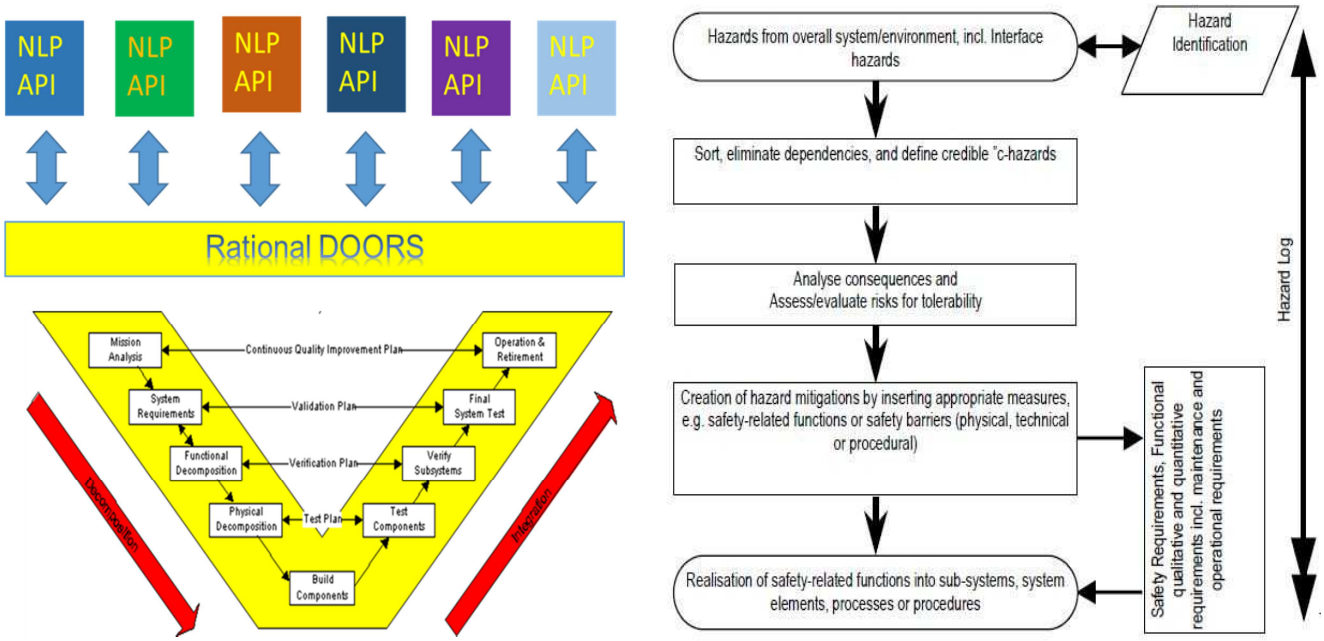

Figure 1. (a) Lifecycle management and watch-dog functionality, (b) Risk assessment process [4]

Figure 1(a). represents the idea of having an intelligent watchdog function during the entire lifecycle phases, using a combination of techniques such as neural networks and decisions trees for example, see Parkinson and Bamford (2016). There are tools workflow tools under development that seek to facilitate the management and control of compliance against safety standards such as EN50216-1 (1999), assuring the lifecycle compliance of developments against the requirements from the standard, Kristen and Althammer (2015). Their particular approach uses the IBM Rational Doors system engineering/ requirements management tool as the backbone of the approach as depicted in Figure 1(a). The idea would be to plug in Application Programming Interfaces (APIs) such as the one we have developed in this paper into this type of architecture.

Some useful research has been done over the years to identify and qualifying good requirements, for example Carlson et al (2104) who sought to qualify the quality of requirements within NASA which describes a set of metrics used to assess

requirement documents. Requirements are deemed to be the description of system behaviors, without specifying how they should be accomplished. The ARM tool combined multiple studies such as text mining, quality analysis and natural language processing (NLP) to devise a toolset consisting of various quality metrics based on the philosophy above.

Also, the Shallow knowledge analysis toolset Sawyer at al (2005) was devised to acquire knowledge about the problem domain and help human experts to develop the early phase of requirements. This study involved performing on text corpus various statistical language engineering techniques. In particular frequency profiling, Part-of-speech analysis, collocation analysis and semantic tagging were combined to extract named entities, domain terms, norms and business rules from a source document.

NLP is a technique that is not widely used in the rail industry, and there is therefore an opportunity to apply it to the critical systems engineering areas such as requirements and hazard log development and management. Previous work by Syeda (2016) examined how NLP could be used to examine accident reports and look for clusters and correlations that could help to ensure that future heightened risk situation might be spotted more easily and this paper builds upon that work. NLP is being applied more widely in other domains such as social media and advertising. The development processes for complex engineering systems offer great scope for its application. For example, if one looks through the railway safety and reliability standards, EN50126 -1 (1999) and the recommended activities at each life-cycle stage it will be seen that nearly all the items are data driven. Using NLP, Big Data and Machine Learning (ML) a lot of the compliance checking activities undertaken by currently safety engineers could be eliminated.

1.2. Railway safety and hazard Logs

Safety is one of the most important aspects in the railway industry and is an integral part of the railway system engineering process. Not only is it necessary, the process of managing and delivering safety is complex and costly, and hazard logs play a vital role in both system engineering and risk assessment. Documents of this type are passed around various departments and are used extensively to manage system risks. For example, hazard logs can originate from accident reports or brainstorming sessions, their main use being to record system-wide or subsystem risks.

Usual practice involves manually analyzing the documents to identify the root causes of the accidents, finding the hazards related to such causes (i.e. defining new or identifying existing conditions that could lead to the accident sequence), ranking the priorities of the hazards, devising safety requirements to help mitigate high risk hazards. Due to the complex nature of railway systems, high levels of interdisciplinary collaboration and system knowledge are required to complete this type of analysis to ensure safety risk is being appropriately managed.

Safety analysis processes involve constantly recording and assessing past accidents, finding the root causes and escalation scenarios. Risk assessments comprising risk analysis, evaluation and acceptance is the process used to address the identification of accidents and hazards, quantification of risks and the judgment on the tolerability of risks [4]. The steps also involve analysis of the causal analysis of hazards. Hazard logs play an important role in this process. Figure 1(b) shows a typical flow chart of the risk assessment process taken from EN50126-2 (2007) which also provides the following definitions used in the process:

- Hazard Log: “the document in which hazards were identified, decisions made, solution adopted and implementation status are recorded and referenced”.
- Hazard: “a “*condition* that could lead to an accident”.
- Cause: the initiator for an accident sequence.
- Consequence: also known as Effect or Potential Accident. These represent unintended events that are harmful to hum, property or environment.
- Mitigations: measures and actions taken to lower hazards to tolerable levels. These could be safety requirements that needed to be implemented in the system.

Section 2 describes some of the data processing techniques used in NLP. Section 3 describes the NLP data and associated methods and the categories of content to assess quality. In section 4 and evaluation of the tool is made and results presented. The discussion in section 5 provides a look at the results in a wider context and makes conclusions, with the directions for further work clearly identified.

2. Machine learning & NLP techniques

The following is a summary of the main techniques that can be used to analyse textural content.

- Text Classifications. Text classification is a subset of supervised learning, which involves using training data that has already been labeled by human supervisors. The goal is to achieve accurate classifications on new text, Manning, Raghavan and Schutz (2008). This technique was applied in various situations that required categorization of a piece of text. For example classifying financial reports into different headers (chairman statements, governance reports, etc.) Alves et al (2016) or sentiment classification of movie reviews, Pang et al (2002). For this project, we used text classification on short, sentence-length texts instead of the whole document text.
- Bag of Words. For machine learning to work on textual data, the text must be represented in the form of numerical features. *Bag of words* is a common method of word featurization, Manning, Raghavan and Schutz (2008). In short, a document is represented by a list of non-ordered frequency, which contains all the *terms* in that document. The frequency could be calculated for single words or a pair of words (bi-gram), three words together (tri-gram) or more (N-gram). By representing a text document by a list of *term* frequency, further machine learning techniques can be applied to the document using this numerical data.
- Term frequency-inverse document frequency. Term frequency (tf) metric is simply computed by calculating the frequency of all words in a document. This is not helpful when we want to compare different documents from the same domain. Two documents about industrial safety would most likely have similar high frequency terms, such as; 'safety', 'hazard', regardless of what industry these documents are related to. In order to reduce the effect of keywords that appear very frequently in the domain, the technique of term frequency-inverse document frequency *(tf-idf)* was introduced [8]. This involves calculating the inverse document frequency, which represents how important a term is in a class of document, or how rare it is across different classes of documents.
- Word Vector. Vector space model takes the analysis a step further, by representing a text document as a data structure, Mikolov et al (2013). This is created by applying a term weight to an extracted document's *bag of words*. One such term weight can be that terms *tf-idf*. In other words, this model represents a document with a *feature vector*, where the features are the terms', Mikolov et al (2013).

3. Data and methods

A list of quality metrics have been compiled that deal with the identification of red flags or green flags from individual cell texts within a hazard log, i.e. finding the bad or good indications from the texts. The overall quality of the document was not assessed. This was due to the differences in models and format standard of the provided data, which made it difficult to generalize a solution for any type of hazard log. For example, the whole-document metric noted in the NASA requirement checking tool ARM was the document readability, Carlson and Laplante (2014). This can be measured by different grade level indices e.g. the Flesch-Kincaid grade level index. Textual data in the hazard logs analyzed here, however, was in the highly technical class, therefore such metrics would not be particularly helpful.
The quality metrics identified in this project are noted in the following sections.

3.1. Lack of safety impact

To focus a risk assessment effort on the most significant hazards, a preliminary hazard analysis should be performed to rank the hazards in order of their risk. To rank the hazards, a safety impact is required.
For this metric, a term frequency-inverse document frequency was performed on the Effect column in the provided datasets. The output was a set of keywords that appeared in the Effect column at a higher frequency than the other columns. The items that represented an accident or safety issue were selected and cross checked with a commonly used standard railway safety guide [4]. These included words such as those shown in Table 1 below:

Table 1: Typical Rail Accidents

Injury	Shock
loss of life	electrocution
collision	burn
fall	crush
trap	

3.2. Multiple clauses

Multiple clauses were flagged red if the cell text included more than one sentence and split by various punctuation marks, such as full-stops.

3.3. Standard list of items

For this item and associated red flag, a standard list of predefined items was used to represent terminology present in safety standards and best-practice risk documentation. Input cell text was compared with exact sentences in this in this list of items. Checking for sentence similarity in terms of semantics, did not help in enforcing a writing style in the documents and so the simpler rule was used. Also, for large documents, a sentence similarity check would significantly slow down the analysis, and would defeat the purpose of providing a quick quality check.

3.4. Mixed up content

To identify texts that were likely to belong to another header, we used cleaned hazard logs as training data. Their columns in the files were used as the 'correct' answer in header classification.
The toolkit "weka" was used for this task. The input texts were first converted to word vectors, Frank et al (2016). The n-gram size was set to 1, and only the top 100 words were kept.
Snowball stemmer was used to reduce the words into their root forms, Porter (2001). This ensured words like 'injuries' and 'injury' have the same weighting during training.
Using Naive Bayes techniques and the above settings, we achieved an accuracy of 90.9% on the training data set. Table 2 showed the accuracy in more detail.

Table 2: Accuracy by class.

Class	Precision	Recall
Cause	0.867	0.845
Effect	0.974	0.968
Hazard	0.829	0.86
Mitigation	0.971	0.967

3.5. Ambiguity

As a proof of concept to check the reliability of this red flag identifier, an Ambiguity classifier was trained based on the experiment 2 results. 2 experiments were conducted using expert to assess randomly chosen cells from the hazard logs. These are fully described by Ngo (2017) but are briefly described in the box below.

In Experiment 1, the experts were asked to label single cells randomly chosen with no context as to whether they were hazards, cause, accidents or mitigations. Mitigations were accurately determined, with hazards being the most difficult to accurately determine, i.e. the consensus between the experts was lower. In Experiment 2, the actual hazard chain was shown with columns, hazard, cause, accident, and mitigation headings all present. One cell was highlighted by the NLP. The expert was required to assess the cell against the following criteria

- Is Ambiguous : as an attempt to gather further opinions on this potential red flag. We wanted to see if the experts can point out ambiguous items, with their whole hazard entries provided.
- Is Lack of details : like above, we want to find cells that are lacking details.
- Is in wrong header : an attempt to measure the agreement on header classification again, now with extra information provided.

As might be expect the second experiment the agreement between experts was high given the increased context. The idea was to look for agreement between the NLP and the expert. This showed that the experts had most problems spotting the ambiguity than the other two criteria. This is probably driven by the different the view points of the experts but will be an interesting subject to further research.

The training data was taken from the manually labeled texts that were either marked “Is Ambiguous” or “Is Lack of detail” by either expert rater. One expert’s idea of Ambiguity may, however, be different from another expert’s view. The idea was to pool different opinions to build a simple ambiguity detector. 127 instances were used to train this classifier. The choice of technique was Naive Bayes which works better with small data sets. Other settings were the same as the Header classifier mentioned above. The overall accuracy was 88.98%. However, the result was not strong, as shown in Table 3. The recall for Ambiguous texts was only 0.5. Although Ambiguity is certainly challenging to spot automatically, it was still included in our analysis.

Table 3: Accuracy by class.

Class	Precision l	Recall
Ambiguous	0.786	0.5
Non-ambiguous	0.903	0.971

3.6. Weak Phrases

A more reliable way to spot Ambiguous cell texts was to check for weak keywords. Part of this list came from the NASA ARM tool, Carlson and Laplante (2014). The ARM tool defined weak phrases as words and phrases that introduced uncertainty to requirement statements, Carlson and Laplante (2014). These keywords are then used to represent some level of ambiguity in the technical documents. As requirement statements are also a part of hazard logs, it made sense to utilize these in the analysis. The list of weak phrases from ARM were added to using some generated by performing term frequency-inverse document frequency (tf-idf) on the texts that were marked as ambiguous by the experts. Certain domain terms and inappropriate phrases were removed, while the rest was added to ARM’s list of weak phrases as shown in Table 4. Weak phrases included:

Table 4: Weak Phrases

Adequate	Normal
As appropriate	Provide for
Be able to	Easy to
Capability to	Incompatible*
Capability of	Inaccurate*
Effective	Proper*
As required	

The last three items with an asterisk ‘*’ were identified from experiment 2 (see section3.5), while the others were referenced from, Carlson and Laplante (2014). An example of keywords which did not make it on to this list include; ‘cctv’ which was simply a domain term that was most likely in the ambiguous texts by chance, and ‘direction of travel’ which got to the top of the tf-idf (term frequency-inverse document frequency) list because multiple items in the data set contained these phrases and they occur with ‘incompatible’ in front of them. Some keywords identified by tf-idf were there just because they co-occurred with the actual weak phrases, rather than being weak themselves. However, only using 127 items in the experiment quite limited the accuracy of this analysis.

3.7. Other quality metrics using Keywords

Beside the identification of weak phrases, the NASA ARM tool also provided other quality indicators tailored for requirement statements. As a hazard mitigation could be treated as a requirement, these metrics were included in the analysis.

- *Imperatives:* command words that stressed the necessity of the requirement, for example: shall, must, etc.
- *Directives:* words that strengthened the mitigation by directing extra information in the cell. These keywords were considered *green flag* in this project.
- *Continuances:* like Directives, these words indicated more detailed specifications following a requirement. Like Directives, the Continuances were ‘good signs’ of a Mitigation cell. Examples include: ‘listed:’, ‘as follows:’.
- *Options:* indicators of a loose specification that should not be used in requirements or mitigations. Examples include: ‘may’, ‘can’.

4. Evaluation and Results

To evaluate the tool, a separate data set not used during the model learning and keyword analysis was acquired. The analysis was performed by the tool and the results exported. The data set had 300 rows, representing the hazard log entries. There are 1200 cells in total, as each row has 4 standard columns that are analyzed by the tool. 325 cells contained missing data; this is a high number and was marked as a red flag. In this hazard log many of them belonged to the Mitigation column, probably indicating this log was likely to be in an early phase of development where most of the hazards identified have not yet been fully analyzed

Figure 2 shows the output charts visualizing the distribution of red flags across the headers. In particular Figure 2a offers a quick overview of the document's quality. Each row in the hazard log had its total count of red flags calculated. The document was then divided into categories depending on the number of red flags a row had. For example, 42% of the rows had 4 or more red flags, which is almost half of the data set, indicated by the red sections in the pie chart. Figure 2c provides a little more insight; by listing all types of red flags found in the document e.g. the Ambiguity category had 108 instances, across the 1200 cells. This chart also shows where the red flags are located, in terms of the column that the cell belongs to. For example, "LackImperatives" is a red flag that only exists in the Mitigation column, while "MultipleClauses" can be found in any column, and is shown as a stacked bar in the chart.

Table 5: Wrong red flags indicated.

Red flag	Identified	Wrong	Percentage of Positive False
All Red Flags	589	26	4.4%
Multiple Clauses	51	12	23.5%
Mixed Header	46	11	23.9%
Has Weak Phrases	109	3	2.75%

The analysis results were then exported as a spreadsheet. This was a filtered version of the original input hazard log, containing only the columns that are believed to indicate the standard headers, and the red flags attached to each row of the file. This was then evaluated by the experts in terms of accuracy of the assessment. The people asked to evaluate the results were not involved in production of any data sets used in this analysis thus avoiding biases in their feedback.
The experts were asked to evaluate each red flag by: checking if it was appropriate for the cell or not, providing feedback on missed red flags and providing additional comments about the analysis result.
In summary, out of the 589 red flags identified, 26 were marked as inappropriate, in other words, they were false positives. Table 5 shows the distribution of the wrongly identified red flags.

Surprisingly, while seemingly a straightforward method of checking and implementation, our "MultipleClauses" red flag identifier contributed a high percentage of false positives. From the feedback comments and checking the actual data points, it was found that many of the instances marked as "Multiple Clauses" were due to extra text that provides the context. Other cases included context sensitive text. For instance, a Cause cell might provide the reason why part of the system broke down, with a high level view. Such cells were marked as having a red flag, while in truth they were acceptable. An example of this is the piece of text 'Error in Train Routing Plan. Chosen route unsuitable for train (planning error)'. While the program flagged multiple sentences as an issue and suggested the user split them up, the cell with this text was fine, and did not need to change.

The "MixedHeader" red flag identifier worked by classifying the cell text's header, then comparing it with the actual column that the cell belongs to. 11 of the 46 instances of this red flag were deemed inappropriate by the marker. In the evaluation data set, the cell was fine and the header does not need to be changed. The marker only pointed out the cells that are in the correct header but was identified as having a red flag. There was no case where both the classified header and the actual header were wrong.

A few "HasWeakPhrases" instances were marked wrong due to the context that those phrases appear in. This is a common issue when working with keywords found by calculating term frequency-inverse document frequency, since it does not capture semantics and co-occurrences, for instance, compound words that are already in the industry terminology. One keyword that was looked for was 'normal', so a cell with 'normal operation' was identified as having a weak phrase. This

was not the case, since 'normal operation' belongs to the vocabulary that is used widely by the experts, so there's no real ambiguity.

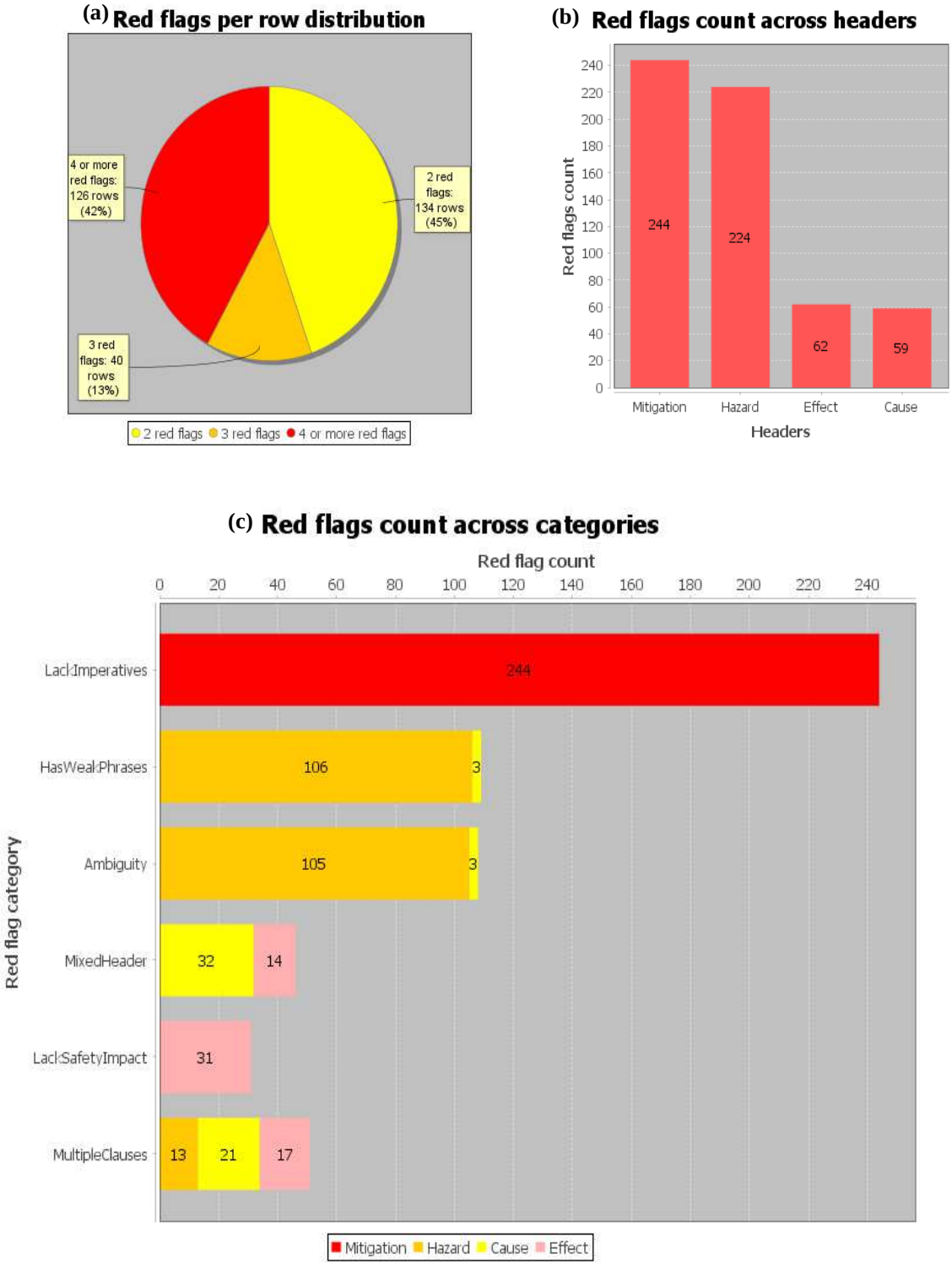

Figure 2: Analysis Dashboard (a) Distribution of red flags count per row (b) Red flags distribution across the headers (c) Red flags distribution across different categories

5. Discussion and Conclusions

This project used railway industry safety documents as data sets for evaluation. Such data is not easily available online or from open source projects. The documents were sourced from a number of projects and contained a range of different

formats, models, standards, and even languages. Even though all three of the provided hazard logs were in excel xlsx format, there were few other similarities in their structures. The files contained a myriad of sheets, headers and layers of information. One difficulty arose from the usage of linked tables.

Understandably, to enforce a writing standard and keep the hazard log organized, one of the files defined a set of standard Causes, Effects and Mitigations, and only used their ID in the actual Hazard log sheet. This is useful for the user to work with these files, but it posed a challenge as it was only used by one of the projects resulting in development of a case specific interface to allow automatic analysis of the text.

Similarly, to this formatting issue, different logs may use different hazard models. For instance, a hazard can be linked to multiple Causes, Effects, and Mitigations; however, one possible model could be that the Causes, Effects and Mitigations are completely unrelated, as shown in Figure 3a. In other words, each entry had the Hazard at its core item, while multiple other cells were just merely related to it. This went against the view that a hazard entry should be related to a whole accident sequence, which contains one Cause, one Effect, and one Mitigation, as illustrated in Figure 3b. The difference being, in Figure 3b the hazard log reader would know which Cause and Effect are linked together and it should form a sensible accident sequence. While in Figure 3a we only know a set of Causes are related to a set of Effects, without a simple way to extract the sequence. Again, this difference made it hard to set a general way to extract information from any hazard log.

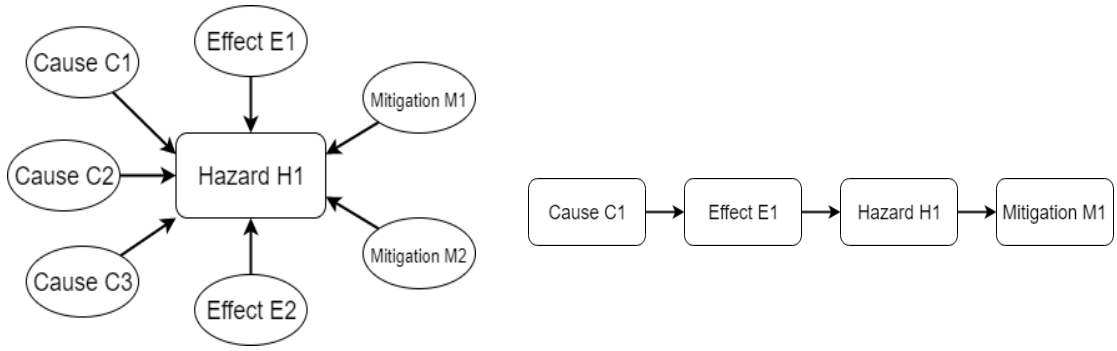

Figure 3: Different hazard log entry models (a) Spider model (b) Line by line model

Given these difficulties, it was assumed that hazard logs in general have a reasonably standardized input. Meaning the input spreadsheet should have only one line at the top denoting the columns in the hazard log. The number of columns, their names and the cell content are not assumed to be standardized. With these assumptions in mind, a tool was built that could quickly provide feedback on the relevant texts within the input hazard log. Using available classification models, the tool could process a large amount of hazard log entries in a short time. e.g. for the sample with 300 rows, it took around 3 seconds to analyze and visualize the results.

The tool identified red flags, as well as where they occur, i.e. which column contains the most red flags, and what categories they are assigned to. Results from the evaluation of a sample hazard log in shown in Table 5, indicated that 563 out of 589 red flags, or 95.6% being correctly identified. We have developed some hazard log quality indicators by combining background research, gathered feedback from the domain expert, and performing data explorations and experiments. These quality indicators have been shown to perform well against a manual evaluation on a sample data set.

In summary, the demonstration software succeeded in quickly providing feedback on an input of hazard log textual data. A wide range of red flags categories could be identified in each text cell of the hazard table with high precision. This would help readers assess the overall quality of the hazard log document, as well as getting an idea on what red flags are prevalent in the document. Such a tool is not known to be used in the railway safety field presently, so it is hoped that this work has provided some additional insight into how hazard log quality can be improved. Even though the tool was not yet integrated and usable in the industry scale, it provided a proof of concept on how such toolset can be developed and applied in the risk assessment process within the railway industry.

Future work will involve refining and extending this tool to deal with application at different levels in the system hierarchy and to enable it to work autonomously as the railway project is in progress. We are also looking at developing

other AI tools that can be plugged into the system lifecycle, for example to autonomously monitoring for heightened risk levels on the operational railway. Another area we are exploring is how NLP and machine learning can be exploited in the providing verification and validation evidence for safety critical systems. The end game is to develop a suite of resilience tools to enable the railway to be safe and more efficient and less wasteful. It was felt during the research that training the NLP to assess data was a glimpse into the future of work in the systems engineering arena.

6. References

Alves, Paulo and El-Haj, Mahmoud and Rayson, Paul and Walker, Martin and Young, Steven, Heterogeneous Narrative Content in Annual Reports Published as PDF Files: Ex- traction, Classification and Incremental Predictive Ability (July 1, 2016).

Carlson N and Laplante P. 2014. The NASA automated requirements measurement tool: a reconstruction. Innov. Syst. Softw. Eng. 10, 2 (June 2014), 77-91.

Frank, E, Mark A. Hall, and Ian H. Witten (2016). The WEKA Workbench. Online Appendix for ”Data Mining: Practical Machine Learning Tools and Techniques”, Morgan Kaufmann, Fourth Edition, 2016.

EN50126-1 (1999)Railway Applications - The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS). Part 1: Generic RAMS Process

EN50126-2 (2007) Railway Applications - The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS). Part 2: Guide to the application of EN 50126-1 for safety

Elliott, Bruce Jeffrey (2014), Benefits of adopting systems engineering approaches in rail projects, Ph.D. thesis, University of Birmingham.

Kristen E, and Althammer , E Published 2015 in SAFECOMP Workshops FlexRay Robustness Testing Contributing to Automated Safety Certification

Manning C D. , Raghavan P. , and Schuetze H. 2008. Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA

Mikolov, Tomas & Chen, Kai & Corrado, G.s & Dean, Jeffrey. (2013). Efficient Estimation of Word Representations in Vector Space. Proceedings of Workshop at ICLR. 2013. .

Ngo, DNT, Exploiting Natural Language Processing and Machine Learning for Analysing Railway Hazard Log, Lancaster University School of Computing and Communications’ MSc Data Science Dissertation September 7, 2017

Pang B., Lee L., and Vaithyanathan S.. Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), chapter Thumbs up? Sentiment Classification using Machine Learning Techniques. 2002.

Parkinson H J, and Bamford G, Big Data and the Virtuous Circle of Railway Digitization, Springer, 20 Oct 2016: Advances in Big Data: Proceedings of the 2nd INNS Conference on Big Data, October 23-25, 2016, Thessaloniki, Greece. http://www.springer.com/gb/book/9783319478975

Sawyer P, Rayson P, and Cosh K. 2005. Shallow Knowledge as an Aid to Deep Under- standing in Early Phase Requirements Engineering. IEEE Trans. Softw. Eng. 31, 11.

Sommerville, I. 1994. ‘Artificial Intelligence and Systems Engineering,’. In Prospects for Artificial Intelligence, ed. A. Sloman, D. Hogg, G. Humphreys, A. Ramsay and D. Partridge, Amsterdam: IOS Press.

Syeda, Kanza Noor and Shirazi, Syed Noor Ul Hassan and Naqvi, Syed Asad Ali and Parkinson, Howard J. and Bamford, Gary (2017) Big Data and Natural Language Processing for Analysing Railway Safety. In: Innovative Applications of Big Data in the Railway Industry. IGI Global Publishing. ISBN 9781522531760