Text Mining and Social Media Mining - Paper Review 1

Construction accident narrative classification: An evaluation of text mining techniques

Author: Michele Guderzo

Warsaw, Academic Year 2021/22

Contents

1 Introduction

The scientific research named *"Construction accident narrative classification: An evaluation of text mining techniques"*, conducted by Yang Miang Goh & C.U. Ubeynarayana in 2017, focuses on the main text mining techniques for the classification of injury narratives in the construction field, evaluating six machine learning algorithms in accordance to 11 accident types.
We are going to analyse this work, highlighting its strengths and weaknesses. However, before doing so, we feel it is appropriate to give the reader an overview of the phenomenon.

2 Paper Review

2.1 Background and Aim

Prevention in the workplace is a topic that has attracted the attention of companies and organisations all over the world, and it is a phenomenon that has been increasingly present in recent years. Despite all the attention that can be paid to prevention, accidents still happen.
This study, starting with the analysis of 4470 publicly available construction accident narratives (1000 labelled, 3470 not labelled), aims to evaluate the utility of various text mining classification techniques in classifying these accidents.
This task, however, is not easy: accident reports are typically unstructured or semi-structured free-text data that require significant manual classification before statistical analyses can be conducted to facilitate interventions. Moreover, an important amount of resources need to be spent on classifying accident narratives, but the classification is not so consistent. On the other hand, organizations that choose not to classify accident narratives will suffer loss of precious data for learning and accident prevention.
To address these issues, companies and organisations have shown increasing interest in automatically classifying and coding incident narratives using text mining techniques, seeking to improve their consistency, productivity and efficiency. It can be argued that, with the increase of these parameters, it is possible to collect more incident data and conduct more detailed analysis to produce useful insights that would not otherwise be available.

2.2 Literature review

One of the common tasks in text mining is classification of text data. Text classification (or categorization) is the task of assigning predefined categories to free-text document, with the aim of helping us to have a broader view on document collections.

Data for text classification is typically represented using vector space model. In this model, each document is represented as a vector of terms. To distinguish between documents in a corpus, each term for each document is given numeric values to show the importance of that term to the document. For this purpose the author used the so-called "term frequency–inverse document frequency" (TF-IDF) representation. Given by $x_{ik} = f_{ik} \times \log\left(\frac{N}{n_i}\right)$, the *TF-IDF* representation is intended to reflect how important a word is to a document in a collection or corpus, where f_{ik} is the frequency of feature i in document k, N is the number of documents in the corpus, and n_i is the number of documents where i occurs. We agree with this choice, since the TF-IDF representation is one of the most used as a weighting factor in searches of information retrieval. Once the document is represented using a suitable vector space representation model, the data can be trained and classified using typical data mining techniques.

To evaluate the performance of the machine learning algorithms experimented, this study adopts the use of recall, precision and F1 score, given by the formulas:

$$Recall = \frac{TP}{(TP + FN)} \tag{1}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{2}$$

$$F1\ Score = \frac{2 \times\ Precision\ \times\ Recall}{Precision\ +\ Recall} \tag{3}$$

Recall indicates the ability of a model to find all the relevant cases within a data set, while precision indicates the ability of a classification model to identify only the relevant data points. On the other hand, F1 score is a harmonic mean of recall and precision, which combines these two measure to provide an overall assessment of performance of the classifier. It is a number ranging from 0 to 1: the closer it is to 1, the more accurate the test.

After, this research focuses on the literature review, saying that there have been other studies in which different text mining techniques were applied to classify injury narratives. It was seen that, in the study by Chen et al. (2015), the best results were produced using matrix factorization coupled with support vector machine (SVM). McKenzie et al. (2010), comparing keyword search, index search, and text mining, found that text mining approach provided the best performance. On the other hand, Berke et al. (2012) showed that the Naïve Bayesian classifier was able to achieve "approximately 90% accuracy" for classifications. Another interesting result was achieved by Tanguy et al. (2015). Using the support vector machine technique, they discovered that, among the different forms of text units, bi-gram and tri-gram of stemmed narratives produced the best results. Furthermore, using Fuzzy and Naive Bayesian models, the algorithms created by Taylor et al. (2014) achieved a sensitivity (equal to recall) between 0.602 and

0.74. Finally, Tixier et al. (2016) used a customized term lexicon (keyword dictionary) along with a set of rules to automatically classify construction incident narratives. It was reported a F1 score of 0.95, which indicates an accuracy of the conducted test close to maximum.
Here the focus is on the fact that, although the successes of machine learning algorithms are evident, these studies deal specifically with emergency department narratives. According to the author, there are not many studies concerning the application of machine learning algorithms in the construction field, reason why they want to focus on it. We give them credit for conducting research in a new field that has not been explored before.

2.3 Dataset

The dataset used by the author is based on accident narratives collected from the US OSHA website (Occupational Safety and Health Administration, 2016). From a dataset of 16,323 accident records, a dataset consisting of 4471 construction industry cases were compiled based on the Industry Code (SIC code).

To prepare the dataset for analysis, the author provided here a pre-processing of the data.
They chose that only the title and accident narrative information were used to classify the cases. As we do not have access to the original database from which the author drew their data, we do not know what other fields were initially present. It would have been interesting to have other variables, first of all the place of the accident, and secondly, the time. Perhaps a field relating to the weather would also have provided some additional information as well. It is our opinion that certain accidents may occur more frequently in certain places or at certain times of the day (e.g. in the evening when workers are more tired and visibility is reduced).
Of the 4471 cases, only 1000 cases were manually labelled following the classifications used in Workplace Safety and Health Institute (2016). Here, the author have tried to merge labels with similar meanings and make them more generic by removing or modifying words within them. We agree with this choice: although information is lost, this made it possible to reduce the number of labels which would otherwise have been too many and too specific. It is a fair compromise between efficiency and distinctiveness.
Afterwards, it can be seen that the actual *Natural Language Processing* (NLP) of the data was conducted: narratives and title were merged into a single paragraph of text, stop words were removed and word stemming was carried out using the Snowball Stemmer.
Here, there is one thing that is not clear: it is not explained why they merged the title and the narratives, but it all depends on how the word 'narratives' is understood. If by this term one were to refer to the entire incident report written by a human classifier, then we agree that title and narrative should be

unified in the same body of text. But if this term were to identify only the story of the accident, then the narrative could loses part of its reliability because an external text is inserted. It is nevertheless true that the title represents an initial summary of what happened, but the 'originality' of the narrative is undermined.

Removal of stop words is fine: *stop words* are insignificant words or phrases that often occur, but they don't provide any meaning and are, therefore, useless for the purpose of the analysis. Generally, stop words are collected in the form of a table creating the so-called "stop-list".

Finally, they shortened each word by stemming. *Stemming* is the process of bringing words to the basic grammatical form, which can be their word stem, base or linguistic root. The goal is to reduce inflectional forms of a word to a common base form, but there could be two main errors: the first one is called "over-stemming", and it occurs when two words of a different stems are stemmed to the same root. The second one is "under-stemming" and, similarly, it occurs when two words are not of different stem but they are stemmed the same root.

The difference between the original data and the processed data is shown in Figure 1.

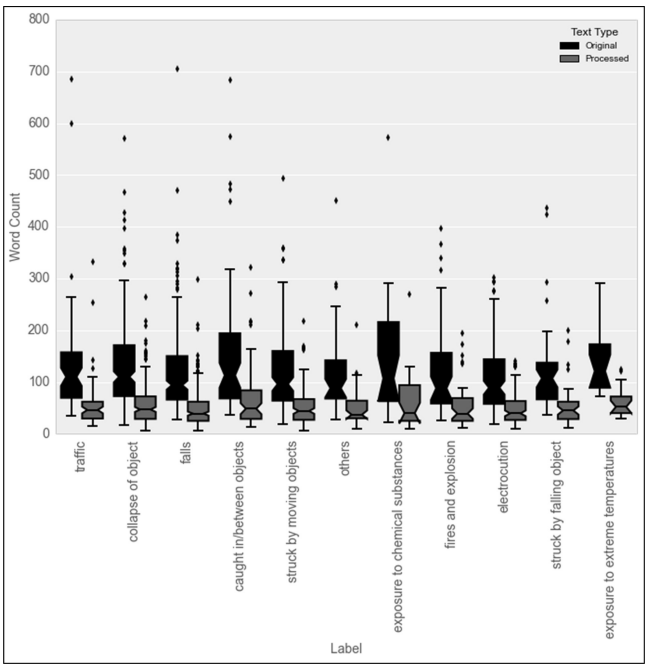

Figure 1: Box plots for word counts - Original data vs Processed data

From 1 it can be seen that the data process was successful: the data was compressed, so the word count was visibly reduced.

2.4 Experiments conducted

Subsequently the author broke each stemmed narrative into strings of N-consecutive words (N-grams) in a process called tokenization. *Tokenization* is the process to identify the smallest text units (mostly words) in a sentence of strings. Called, in fact, "token", they are used as the input data in text mining. There may be different types of tokens: "uni-gram" is a token with only one word, "bi-gram" indicates a token with two words and, similarly, "tri-gram" is a token with three words. Thanks to the tokenization, it is possible to turn a collection of text documents into a set of numerical feature vectors (matrix), and this process is called "vectorization".

In this study, uni-grams and bi-grams were converted into the TF-IDF matrix, which, together with the case labels, was then used to train the classifiers.

Out of 1000 labelled cases, the author, using stratified sampling, set aside only 251 cases to compare the performance of the different approaches evaluated. Is not much, but it is still a sufficient number.

Then, they talk about the six machine learning classifiers that were chosen to conduct a preliminary experiment of multinomial classification. The classifiers, specifically, are: *support vector machine* (SVM), *linear regression* (LR), *random forest* (RF), *k-nearest neighbor* (KNN), *decision tree* (DT) and *Naive Bayes* (NB), which are brefly described. Although there are other text categorization classifiers (such as Boosting and Latent Semantic indexing), we agree with the use and comparison of these six, since all of them are well-known classifiers in text mining. However, we think it would have been interesting to have deepened and analysed the data using the neural networks model. Loosely inspired by the simplification of a biological neural network, this computational model relies on training data to learn and improve its accuracy over time. But once the algorithm is set up for accuracy, it becomes a powerful tool that allow us to classify and cluster data at a high velocity.

As could be seen in other studies, the best performing classifier was the SVM. The decision the author made was to set up the initial SVM was as a One-vs-Rest (OVR) linear classifier with parameters C = 10 and “balanced” class weight.

In the next section we are going to analyse the results and the findings the author have made, paying attention to the comparison of the different machine learning classifiers.

2.5 Results and findings

This section begins with a first preliminary classification using the 6 machine learning algorithms, in order to select the one that provide the best performance. The results are shown in Figure 2.

In Figure 2 author showed the F1 score for all the six algorithms, giving for each the average of all the eleven labels. It can be seen that the best performer is, overall, the SVM. Interesting enough, in the first

	F1 Score					
Labels	**SVM**	**LR**	**RF**	**KNN**	**DT**	**NB**
caught in/between objects	0.60	0.36	0.53	0.34	0.46	***0.89***
collapse of object	***0.66***	0.48	0.54	0.52	0.56	0.25
electrocution	***0.95***	0.91	***0.95***	0.69	0.92	0.67
exposure to chemical substances	***0.62***	0.00	0.00	0.40	0.40	0.40
exposure to extreme temperatures	***0.67***	0.00	0.00	0.40	0.25	0.00
falls	***0.78***	0.63	0.74	0.66	0.76	0.17
fires and explosion	***0.74***	0.56	0.71	0.50	0.64	0.63
others	0.43	0.00	0.29	0.20	***0.48***	0.00
struck by falling object	*0.14*	0.00	0.00	0.11	0.27	***0.61***
struck by moving objects	***0.58***	0.48	0.45	0.44	0.55	0.48
traffic	***0.67***	0.48	0.40	0.54	0.54	0.36
Average	***0.62***	0.35	0.42	0.44	0.53	0.41

Figure 2: F1 scores for preliminary experiment

and ninth label, the outputs of the Naive Bayesian algorithm outclass the ones of the SVM, and not by a little, even if, on average, the NB classifier's performance is among the last positions.

Once the best was found, the study proceeded to optimise its performance through different combinations of tokens, finding that uni-gram offered the best results. Here the author found that the RBF (Radial Base Function) SVM algorithm, with uni-gram tokenization, offers on average the same (high) performance as the linear SVM algorithm. However, the latter is preferred as it is the simplest classifier. We agree with this choice: for the same performance, the simplest solution is the one to lean towards, not only because it is easier to implement, but also because it requires less computational calculation by the computer.
Finally, the values of precision, recall and F1 score are shown for the SVM model.

In the last part of this section, the author focus on the topic of misclassification and its probable causes. The first cause was identified with the TF-IDF matrix, as it is not able to capture the context in which the words were used in the incident. Context understanding is only solved to a small extent with the use of the two-gram and tri-gram, however the best performance was seen with the use of the uni-gram. The use of two- and three-grams makes sense when combinations of several words constitute a meaning of their own, but as we have seen this is not the case. On the contrary, using more words increases specificity and risks giving too much importance to semantically useless words. Other causes were found in the difficulty of classifying certain labels and, consequently, in the intrinsic ambiguity that characterises natural

language: since the same concept can be expressed in several ways, (through, for example, synonyms) human classifiers can report two different labels with the same, or very similar, semantic validity.

3 Conclusions

We reviewed the scientific research of Yang Miang Goh regarding the evaluation of text mining techniques.

Overall we can say that this paper is quite complete and detailed, it treats each topic in depth and provides adequate explanations for questions that may arise during the reading process.

In spite of the problems encountered in the field of misclassification, to which various solutions were put forward, the research was carried out critically in all its aspects.

With regard to future work, in addition to that already mentioned by the author in the appropriate section, we recommend evaluating a technique similar to Latent Semantic Analysis (LSA), namely Latent Dirichelet Analysis (LDA). This well-known algorithm would allow to observe how words and phrases co-occur in documents in order to group the words that best represent them, providing a coherent topic.