Safety Science 89 (2016) 72–76

Contents lists available at ScienceDirect

Safety Science

journal homepage: www.elsevier.com/locate/ssci

Visual analytics for text-based railway incident reports

Miguel Figueres-Esteban *, Peter Hughes, Coen van Gulijk

University of Huddersfield, Institute of Railway Research, Queensgate, Huddersfield, UK

ARTICLE INFO

Article history:
Received 25 November 2015
Received in revised form 23 May 2016
Accepted 23 May 2016

Keywords:
Close call
Visual analytics
Railway safety
Risk analysis
Network text analysis

ABSTRACT

The GB railways collect about 150,000 text-based records each year on potentially dangerous events and the numbers are on the increase in the Close Call System. The huge volume of text requires considerable human effort to its interpretation. This work focuses on visual text analysis techniques of Close Call records to extract safety lessons more quickly and efficiently. This paper treats basic steps for visual text analysis based on an evaluation test using a pre-constructed test set of 150 Close Call records for "Trespass", "Slip/Trip hazards on site" and "Level crossing". The results demonstrate that visual text analysis can be used to identify the risks in a small-scale test set but differences in language use by different cohorts of people interferes with straightforward risk identification in larger sets. This work paves the way to machine-assisted interpretation of text-based safety records which can speed up risk identification in a large corpus of text. It also demonstrates how new possibilities open up to develop interactive visualisations tools that allow data analysts to use text analysis techniques for risk analysis.

1. Introduction

The benefits of analysing Close Call/near misses reports have been proved in many industries (Bliss et al., 2014; Gnoni and Lettera, 2012; Macrae, 2014). In the GB Railways, two systems are in operation today to exploit these benefits: in the Close Call System (CCS) and the Confidential Incident Reporting and Analysis System (CIRAS).

The Close Call System collects about 150,000 text-based records each year on potentially dangerous events and the numbers are on the increase. The huge volume of text from Close Call records requires considerable human effort and time to its interpretation. Computer-assisted Text Analysis (TA) provides alternative techniques that can facilitate the extraction of safety knowledge and reduce the human effort. Three fundamentally different approaches can be found for TA: thematic, semantic and networks (Popping, 2000). Network analysis is in the emerging field of Visual Analytics (VA). VA combine automated data analysis techniques from massive, inconsistent and conflicting data with human knowledge by means of interactive visualisations for an effective understanding, reasoning and decision making (Keim et al., 2010, 2008; Thomas and Cook, 2005). This paper describes the initial steps for using VA techniques and demonstrates a way forward to develop interactive visualisations tools but also demonstrates some of the difficulties on the way ahead.

It is beyond the scope of this paper to analyse the benefits of different methods that can be used for text analysis (e.g. see Popping (2000) for overview). This paper uses the analysis method proposed by Paranyushkin (2011). It demonstrates the benefits of a method for representing normalised text as a graph and using network analysis for detecting contextual clusters and key concepts that are junctions for meaning within a text. This VA approach suits the aim of this work. It is used to support interpretation of large amounts of text by graphical representation techniques that reduce the analyst's workload (Crow et al., 1994). The method is based on visual text analysis by means of graphs: terms (words and multi words) are nodes, and their relationships are links in word based graph networks (Drieger, 2013; Paranyushkin, 2011; Popping, 2003). This way of working allows analysis of the type and strength of relationships between the main concepts from a text, and thus, allows information extraction from the graph. To date, no references about using this technique in safety science were found.

2. Methodology

Although it is desirable to analyse all Close Calls in one go, we believe that there are many obstacles that have to be addressed before this is possible. This paper explores the basic principles by analysing a sample of Close Call records that describes three risk scenarios in order to identify them. A pre-constructed dataset of 150 records was constructed by selecting the first 50 records from the Close Call database classified as "Trespass", "Slip/Trip hazards

* Corresponding author.
E-mail address: m.figueres@hud.ac.uk (M. Figueres-Esteban).

http://dx.doi.org/10.1016/j.ssci.2016.05.009
0925-7535/

Table 1
Example of cleaned text, tagged text and tokenised text. The latter is used for the analysis.

Cleaned record
Emailed report from LOM Date: 08/09/13 Time: 1900 ELR: LEN3 59m 14ch Issue – Trespasser on the line in the Hartburn Junction area. Trains cautioned, reported all clear by MOM @ 1930 Action – Fencing to be checked 09/09/13 DU: Newcastle
Cleaned and tagged record in lowercase
Emailed report from local operation manager date _date_ time _time_ elr_code distance_tag issue trespasser on the railway line in the geo_place junction area trains cautioned reported all clear by mobile operations manager _time_ action fencing to be checked _date_ geo_place
Cleaned, tagged and tokenised record without stopwords and in lowercase
Email report from local operate_ manager_ date _date_ time _time_ elr_code distance_tag issue trespasser on railway_ line_ in geo_place junction_ area_ train_ warning_ reported all clear_ by mobile_operations_manager_ _time_ action fence_ check_ _date_ geo_place

on site" and "Level crossing". These records were cleaned of non-desired characters using the NLTK toolkit in Python (Bird et al., 2009) in order to generate the text source to process (cleaned record in Table 1). The "*tagging process*" and "*tokenisation process*" described in Hughes et al. (2015) was used to create the two sets of text for visualising. The visual analysis of the tagged-text (cleaned and tagged record in lowercase in Table 1) provided information to tailor the tokenisation process (removing main stopwords and stemming plurals or verbs), avoiding obscuring main concepts in the tokenised-text network (cleaned, tagged and tokenised record without stopwords in Table 1).

The final tokenised text is composed of terms that are (1) tags related to places, codes or measured entities (i.e. *geo_place*, *elr_code* and *distance_tag*, respectively), (2) tokens that link relevant adjacent words or represent stem verbs and nouns (e.g. *mobile_operations_manager_*, *check_* or *junction_*) and (3) words from the original text (e.g. *trespasser*).

The final text can be transformed into a network building its adjacency matrix of words (aka word by word co-occurrence matrix). An adjacency matrix shows how the nodes of a graph are connected into pair of nodes and it is the input of visualisation tools. In the evaluation test, the adjacency matrix to visualise is the addition of two matrices: one for a context window of size two and one for a context window of size five. The two-gap context window identifies relevant adjacent words such as *access* and *gate* (Fig. 1.2). The five-gap context window takes into account the proximity of the words that are slightly further apart such as *press* and *button* (the sequence would be *press stop_ button*, Fig. 1.2) but it also amplifies the adjacent words by double counting. Gephi software was the visualisation tool selected for the visual representation of the adjacency matrix. The visualisation was made using the Force Atlas layout with the parameters *Inertia* = 0.1, *Repulsion* = 10,000, *Attraction strength* = 10, *Maximum displacement* = 10, *Autoslab Strength* = 80, *Autoslab sensibility* = 0.2.

In order to gather knowledge from the networks two key centrality measures were analysed, the *degree of a node* and the *betweenness of nodes*. The degree of a node is the number of links connecting a node (Lewis, 2011; Newman, 2010). It is represented by the size of the node in Fig. 1 and is an indicator of the importance of the node (for instance *cross_* in Fig. 1.1 or *barrier_* in Fig. 1.2). The betweenness of nodes is defined by Freeman (1978) as the frequency with which a node falls between pairs of other nodes on the shortest paths connecting them (like the *stop_* in the *press stop_ button* sequence in Fig. 1.2). In the text analysis context, the betweenness gives information about the nodes that connect clusters (Paranyushkin, 2011; Popping, 2000). Thus it provides information about the overlap of clusters as shown in Fig. 2. Although the betweennes cannot be expressed in the Fig. 1, the strongest betweenness is considered in the cluster interpretation.

The Louvain method for community detection was applied to detect clusters in the text network. A resolution of 1.5 was given in order to discover large clusters (Blondel et al., 2008).

3. Results

The resulting text network is an undirected graph of 775 nodes and 16,563 edges. The Louvain method identified four clusters with a modularity of 0.611 (Fig. 1).

The first, second and third clusters have the highest degree nodes with a high betweeness (*cross_, geo_place, distance_tag, location, barrier_, access_*, gate_ and *road_vehicle_*) and contain a great quantity of high and medium degree nodes related to level crossings (*elr_code*, *level_crossing, road, driver_, red_, light_, flash_, warning_, miss_, padlock_, unsecure, point, track, trackside_, lock, open_, enter, safe_* or *authorised*). These three clusters present differences regarding the nodes that represent people and the topics that the higher degree nodes describe. The first cluster encloses nodes related to technical staff (for example *network_rail_, operative or signaller)* and operational railway terms such as *box_, signal_, cctv_, elr_code, cess, main_, delay_, safe_, line_, dn_, up_, platform_, bridge_, station_*or *downside*. The second cluster contains two high degree nodes with high weight that describe the general public (*member_, public_* or *pedestrian_*) and diverse road safety terms such as *road_vehicle, barrier_, light_, red_, descend_, stop_, button*, *press, pass_* or *stopped_*. As with the first cluster, the third cluster shows many nodes related to technical staff (for example *mobile_operations_manager, operational, telecommunications, manager* or *engineer*) and operational work terms such as *close_, call_, access_, gate_, miss_, padlock_, unsecure, point, track, trackside_, lock, open_, enter, control_* or *authorised.*

The fourth cluster displays high degree nodes for example *hazard_, potential_, trespass_ or sliptripfall_*, nodes related to people like *worker_* or *member_of_staff* and terms related to the workforce environment such as *tool_, gap_, wall_, sticking, cable_, fence_, boundary_, overgrown_* or *vegetation_*.

4. Discussion

Four clusters were found from the five-word gap tokenised network using a resolution of 1.5. The choice of the resolution influences how many clusters are determined by the Louvain method for community detection. As a guideline, it is accepted that small values (less than 1) generates too many small clusters to extract sensible learning from the data. A value greater than 1 means fewer clusters are created but they tend to be larger in the sense that there are more nodes in a cluster. As we are interested in identifying three risk clusters, a value of 1.5 was used.

The resulting clusters have a modularity of 0.611. According to Paranyushkin (2011) this is higher than the threshold value of 0.4 to indicate stable clusters. Stable means that this is an allowed use of the modularity algorithm. De facto, the connectivity of the terms within a cluster is higher than with other clusters in the network.

The graphs still show some words that could be considered stopwords (e.g. *that, could* or *which*). This is a shortcoming of the method used in this paper. The words identified in the graphs could be re-evaluated and made part of the text cleaning rules in