A BERT-Based Model for Question Answering on Construction Incident Reports

Hebatallah A. Mohamed Hassan(✉), Elisa Marengo, and Werner Nutt

Faculty of Computer Science, Free University of Bozen-Bolzano, Bolzano, Italy
{hebatallah.mohamed,elisa.marengo,werner.nutt}@unibz.it

Abstract. Construction sites are among the most hazardous workplaces. To reduce accidents, it is required to identify risky situations beforehand, and to describe which countermeasures to put in place. In this paper, we investigate possible techniques to support the identification of risky activities and potential hazards associated with those activities. More precisely, we propose a method for classifying injury narratives based on different attributes, such as *work activity*, *injury type*, and *injury severity*. We formulate our problem as a Question Answering (QA) task by fine-tuning BERT sentence-pair classification model, and we achieve state-of-the-art results on a dataset obtained from the Occupational Safety and Health Administration (OSHA). In addition, we propose a method for identifying potential hazardous items using a model-agnostic technique.

Keywords: Hazard identification · Question answering · BERT · Model-agnostic interpretability

1 Introduction

According to the latest fatal work injury rates reported by the International Labour Organization (ILO), construction sites are the most hazardous workplaces [14]. A standard method for identifying hazards in the production industries is the Job Hazard Analysis (JHA). It consists of identifying the work activity, identifying potential hazards related to those work activities, and proposing procedures to eliminate, reduce or control each of the hazards.

In this paper, we investigate possible techniques to support the identification of the risky activities and the identification of hazards related to those work activities. The idea is to leverage on existing data on past dangerous situations or on injury reports to extract such information. Injury reports produced by workers are typically unstructured or semi-structured free-text data, which traditionally relies on human oversight to extract actionable information. Most of the existing works formalize the task of automatic narrative classification as a standard text classification task which consists of two steps: *text feature extraction* and *classification*, with an underlying assumption that the entire text has

P. Rosso et al. (Eds.): NLDB 2022, LNCS 13286, pp. 215–223, 2022.
https://doi.org/10.1007/978-3-031-08473-7_20

an overall topic. However, injury reports in construction typically contain different topics or aspects, such as: *work activity*, *incident type*, *injury type*, and *injury severity*.

Inspired by the recent trend of formalizing different Natural Language Processing (NLP) problems as a Question Answering (QA) task [5,12], we transform the injury narrative classification into a sentence-pair classification task, where the input to the classification model consists of question and narrative pairs. The questions are formulated based on different aspects, such as *work activity*, *incident type*, *injury type*, or *injury severity*. The idea is that the incorporation of aspects forces the classification model to attend to the part of the narrative related to that aspect, and therefore enhances the classification performance. Moreover, we identify potential hazards by extracting the predictive words from the narratives that are most informative for *incident type* classification (e.g. narratives classified to 'fall' if 'scaffold' hazard presents), using the Local Interpretable Model-agnostic Explanations (LIME) technique.

2 Related Work

Several approaches have been proposed for extracting precursors from injury reports based on an entirely hand-written lexicon and set of rules [1,15,16]. Hand crafting of rules has the advantage of being accurate. However, the rule creation process is resource intensive, both in terms of time and human input.

A wide variety of classical machine learning techniques have been employed for classifying injury narratives. For example, [3] proposed an unsupervised approach using TF-IDF and K-Means to cluster injury narratives. [6] evaluated different supervised techniques, and found that SVM produces the best performance. The authors further presented an ensemble approach for construction injury narrative classification [18]. Similarly, [20] proposed an ensemble model to classify injury narratives, and a rule based chunker approach is explored to identify the common objects which cause the accidents. A significant drawback, however, of the TF-IDF is that it ignores the semantics of words.

Recently, there are few works that exploited deep learning techniques. For example, [2] utilised Convolutional Neural Networks (CNN) and Hierarchical Attention Networks (HAN) to classify injury narratives, where for each model, a method is proposed to identify (after training) the textual patterns that are the most predictive of each safety outcome. In [7,21], the word embedding of Bidirectional Encoder Representations from Transformers (BERT) base model is used to model accident narratives. However, to the best of our knowledge, fine-tuning BERT for QA has not been investigated for this task.

3 Dataset

The dataset[1] used in this study is collected from the Occupational Safety and Health Organization (OSHA) website.[2] It has been released by [17] to be used

[1] https://github.com/Tixierae/WECD/blob/master/classification_data_set.csv.

[2] https://www.osha.gov/pls/imis/accidentsearch.html.

as a benchmark for construction injury report classification. The dataset contains 5,845 injury cases, where each case is annotated with different information, including: (1) *identification number*, (2) *narrative*, (3) *cause/work activity* (the activity the worker was involved in before the accident), (4) *fatCause/incident type* (what is the accident, e.g., 'Fall'), (5) *injury type* (the injury nature, e.g. 'Fracture'), (6) *injury severity* (the worker has died, hospitalized, or non-hospitalized).

As shown in Table 1, a narrative is a short text that provides a complete description of the accident. It includes events that led to the accident and causal factors, such as *work activity* and *incident type*. It also states the outcome from the accident, such as *injury type* and *injury severity*. As a preprocessing step, we remove dates and special characters from the narratives using the NLTK[3] Python library. The average length of a narrative is 104 words after the preprocessing step.

Table 1. Sample accident report from OSHA dataset.

Narrative	'On April 9, 2013, Employee #1 was **installing vinyl sidings** on a single story residence. The employee was standing an A-frame ladder that was set on a plank of a scaffold. The scaffold moved causing to lose his balance. The employee **fell** from the ladder approximately 12-ft to the ground. Employee #1 was transported to an area hospital, where he was treated for an abdominal **fracture**. The employee remained **hospitalized**.'
Activity	'Exterior carpentry'
Incident type	'Fall'
Injury type	'Fracture'
Injury severity	'Hospitalized'

4 BERT for Question Answering

4.1 Methodology

Given an input narrative text $x = x_1, \ldots, x_L$, where L denotes the length of the text x. We need to classify x with a label $y \in Y$. Each label y is associated with a natural language description $q_y = q_{y1}, \ldots, q_{yM}$, where M denotes the length of the label description q_y.

We consider our task as a sentence-pair classification problem by generating a set of (NARRATIVE, ASPECT + LABEL DESCRIPTION) pairs, with new binary labels $\in \{yes, no\}$, indicating whether a label should be assigned to the narrative or not with respect to a given aspect. ASPECT + LABEL DESCRIPTION, we name it $\hat{q}_y$, represents the aspect concatenated with the ground truth label to form a question, such as "Is the *work activity* of the narrative *excavating*?" or "Is the *severity* of the narrative *hospitalized*?".

[3] http://www.nltk.org.

We fine-tune BERT sentence-pair model [4]. Thus, we concatenate the label description $\hat{q_y}$ with the narrative text x to generate $\{[\text{CLS}]; \hat{q_y}; [SEP]; x\}$, where [CLS] and [SEP] are special tokens. The concatenated sequence is fed to multi-layer transformers in BERT, from which we obtain the final hidden vector $\mathbb{C} \in \mathbb{R}^H$ corresponding to the first input token ([CLS]) as the aggregate representation. Then, we add a classification layer with weight matrix $W \in \mathbb{R}^{K\times H}$, where K is the number of labels (two in our case). We compute a standard classification loss by a softmax function $f = \text{softmax}\left(CW^{\text{T}}\right)$. We then consider the label description that generates the highest probability for '*yes*' when concatenated with the narrative, as the predicted label of that narrative.

4.2 Baselines

We use the following models as baselines:

- **FastText**: A word embedding model[4] that uses a character level n-gram, which makes it capable of generating embeddings for out-of-vocabulary words [8]. Once the embeddings are obtained, a max-pooling operation is applied, followed by a softmax function to derive label predictions.
- **Convolutional Neural Networks (CNN)**: A classic baseline for text classification [9]. It applies CNN based on FastText pre-trained word embedding.
- **Hierarchical Attention Networks (HAN)**: This method deals with the problem of classifying long documents by modeling attention at each level of the document structure, i.e. words and sentences [19]. This allows the model to first put attention on word encoder outputs, in a sentence, and then on the sentence encoder outputs to classify a document.
- **BERT-base**: We use the BERT-base model [4] and we follow the standard classification setup in BERT, in which the embedding is fed to a softmax layer to output the probability of a label being assigned to an instance.

4.3 Experimental Setup

We split the dataset into a training and a testing set of 80% and 20%, respectively. For HAN baseline, similar to [2], we set the maximum length of a sentence to 50 words, and the maximum number of sentences in a document to 14 sentences. While for CNN baseline, the hyperparameters are set as follows: filter size = 5; number of filters = 128, similar to [22]. For BERT-base and BERT-QA models, we use the pytorch-transformers[5] library, and the uncased version of the pre-trained BERT-Base[6] model. We fine-tune the models for 3 epochs to minimize the negative log-likelihood of predicting the correct labels of the narratives in the training set, using stochastic gradient descent with the Adam [10] optimizer, an initial learning rate of 3e−5 [13], and batch size of 6. Finally, we run our experiments on NVIDIA Tesla K80 GPU with 12 GB of RAM.

[4] https://github.com/amaiya/ktrain.

[5] https://github.com/huggingface/transformers.

[6] https://storage.googleapis.com/bert_models/2018_10_18/uncased_L-12_H-768_A-12.zip.

4.4 Results

In Table 2, we present the performance of our fine-tuned BERT model (BERT-QA) and the baseline models, in terms of macro-averaged precision, recall and F1-score. We observe substantial better performance of BERT-QA in general over the other models. More precisely, the QA strategy using BERT sentence-pair model outperforms the classical BERT classification model. It achieves a performance gain of +2.0% in terms of F1-score for classifying narratives based on *work activity*, +2.0% for *incident type*, +3.0% for *injury type* and +3.0% for *severity*. This means that the incorporation of aspects and label description gives the model the ability to attend to the relevant text in the narratives.

However, the classification based on *work activity* still suffers from poor performance in general, since there are many labels that represent activities which are practically very close to one another (e.g., excavating and trenching).

Table 2. Precision (Prec), recall (Rec) and F1 of the classification models.

Model		Activity	Incident type	Injury type	Severity
FastText	Prec	0.58	0.63	0.75	0.82
FastText	Rec	0.56	0.63	0.73	0.78
FastText	F1	0.56	0.63	0.74	0.80
CNN	Prec	0.62	0.77	0.76	0.89
CNN	Rec	0.54	0.75	0.73	0.79
CNN	F1	0.55	0.75	0.74	0.82
HAN	Prec	0.64	0.71	0.71	0.84
HAN	Rec	0.49	0.73	0.75	0.89
HAN	F1	0.50	0.71	0.72	0.86
BERT-base	Prec	0.62	0.83	0.79	0.90
BERT-base	Rec	0.61	0.82	0.78	0.86
BERT-base	F1	0.61	0.82	0.79	0.87
BERT-QA	Prec	**0.66**	**0.86**	**0.82**	**0.91**
BERT-QA	Rec	**0.63**	**0.82**	**0.79**	**0.91**
BERT-QA	F1	**0.63**	**0.84**	**0.81**	**0.91**

5 Model-Agnostic Interpretability for Identifying Hazards

In this section, we propose a method to automatically extract words related to potential hazards, based on the explanation of *incident type* classification, using the fine-tuned BERT model. More precisely, we automatically extract the parts of the narratives that influence the correct prediction of *incident type* using LIME [11]. LIME is a technique used to explain predictions of any complex or

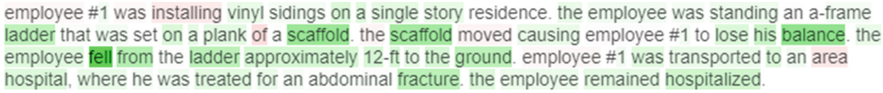
Table 3. Examples of the extracted hazards per *incident type*

Fall			
Ladder	Rope	Scaffold	Sludge pond
Rung	Heart attack	Walkway	Elevator
Struck by falling object			
Falling tree	Hammer	Pipe fell	Tunnel fell
Falling wood	Load fell	Rods fell	Assembly broken
Struck by moving object			
Backhoe slid	Roller overturned	Vehicle	Securing pins
Compactor	Truck	Fall protection	Asphalt roller
Collapse of structure			
Bridge	Columns	Rot	Prefabricated wood
Not designed	Roof collapsed	Falling debris	Collapsed covering
Falling deck			
Electrocution			
Backhoe contacted	Power line	Wire contacted	Fuse
Unprotected conductor	Halogen	Transformer	High voltage
Fire/explosion			
Acetylene	Natural gas	Combustible liquid	Torch
Kettle pot	Unauthorized personnel		
Exposure to extreme temperatures			
Cold	Hot	Humid	Overheated
Steam	Sunlight		
Exposure to chemical substance			
Sulfide	Carbon	Methane	Monoxide
Kerosene	Gas	Bacterial	Hydrogen

non-linear classification model by approximating the underlying model by an interpretable linear model, learned on perturbations of the original instance (i.e. removing words), and then uses the weights of the linear model to determine feature importance scores. In other words, LIME ensures both interpretability and local fidelity by minimising how unfaithful is the local approximation of the surrogate model, g, to the complex classifier, f. The explanation, R, produced by LIME is obtained by the following equation:

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \mathcal{L}\left(f, g, \pi_x\right) + \Omega(g) \tag{1}$$

where x refers to the instance being explained, G denotes a class of potentially interpretable models, $L(f, g, \pi_x)$ is the fidelity function, measuring the reliability of the approximation provided by the interpretable model in the vicinity defined by π_x, and $\Omega(g)$, denotes the complexity of the interpretable model.



employee #1 was installing vinyl sidings on a single story residence. the employee was standing an a-frame ladder that was set on a plank of a scaffold. the scaffold moved causing employee #1 to lose his balance. the employee fell from the ladder approximately 12-ft to the ground. employee #1 was transported to an area hospital, where he was treated for an abdominal fracture. the employee remained hospitalized.

Fig. 1. Explanation of a narrative classified as 'Fall'.

Figure 1 shows an example of LIME visualization for a narrative that is correctly classified as 'Fall', where the most predictive words are 'fell', 'balance', 'scaffold', and 'ladder'. From this we can consider 'scaffold' and 'ladder' as hazardous items. Table 3 shows some examples for the hazards identified for each *incident type*.

Even though the proposed solution does not guarantee to retrieve all possible hazards from the narratives, since it is not simple or straightforward enough to determine the exact source of those accidents, it will help in identifying potential hazards which can then be validated by safety managers. After identifying the potential hazards, we could also produce useful insights about the association between work activities and hazards. For example, most of the injury narratives containing 'scaffold' are related to 'exterior carpentry' *work activity*. We can also get insights about the severity of the injuries when a certain hazard presents. For example, the *injury severity* is 'Hospitalized' and the *injury type* is 'Fracture' for most of the narratives that includes 'scaffold' and related to 'exterior carpentry'.

6 Conclusion

In this paper, we formalize the classification of construction injury narratives as a question answering task. We fine-tune BERT sentence-pair classification model, and we achieve state-of-the-art performance on OSHA dataset. Additionally, we present a method for automatically extracting hazardous items from text based on model-agnostic explanation technique. As a future work, we will expand the questions with synonyms in order to make them more descriptive. Additionally, in the context of a research project COCkPiT, we are developing a tool to assist project managers in scheduling the activities to be performed on-site. We will extend such a tool with a functionality able to highlights the risky activities.

References

1. Baker, H., Hallowell, M.R., Tixier, A.J.P.: AI-based prediction of independent construction safety outcomes from universal attributes. Autom. Constr. **118**, 103146 (2020)

2. Baker, H., Hallowell, M.R., Tixier, A.J.P.: Automatically learning construction injury precursors from text. Autom. Constr. **118**, 103145 (2020)
3. Chokor, A., Naganathan, H., Chong, W.K., El Asmar, M.: Analyzing Arizona OSHA injury reports using unsupervised machine learning. Procedia Eng. **145**, 1588–1593 (2016)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1 (2019)
5. Gardner, M., Berant, J., Hajishirzi, H., Talmor, A., Min, S.: Question answering is a format; when is it useful? arXiv preprint arXiv:1909.11291 (2019)
6. Goh, Y.M., Ubeynarayana, C.: Construction accident narrative classification: an evaluation of text mining techniques. Accid. Anal. Prev. **108**, 122–130 (2017)
7. Goldberg, D.M.: Characterizing accident narratives with word embeddings: improving accuracy, richness, and generalizability. J. Safety Res. **80**, 441–455 (2022)
8. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. CoRR arXiv:1607.01759 (2016)
9. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
10. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
11. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should I trust you?" Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016)
12. Sun, C., Huang, L., Qiu, X.: Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. arXiv preprint arXiv:1903.09588 (2019)
13. Sun, C., Qiu, X., Xu, Y., Huang, X.: How to fine-tune BERT for text classification? In: Sun, M., Huang, X., Ji, H., Liu, Z., Liu, Y. (eds.) CCL 2019. LNCS (LNAI), vol. 11856, pp. 194–206. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32381-3_16
14. Takala, J.: Burden of injury due to occupational exposures. In: Handbook of Disability. Work and Health, pp. 1–22. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-75381-2_5-1
15. Tixier, A.J.P., Hallowell, M.R., Rajagopalan, B., Bowman, D.: Application of machine learning to construction injury prediction. Autom. Constr. **69**, 102–114 (2016)
16. Tixier, A.J.P., Hallowell, M.R., Rajagopalan, B., Bowman, D.: Automated content analysis for construction safety: a natural language processing system to extract precursors and outcomes from unstructured injury reports. Autom. Constr. **62**, 45–56 (2016)
17. Tixier, A.J.P., Vazirgiannis, M., Hallowell, M.R.: Word embeddings for the construction domain. CoRR arXiv:1610.09333 (2016)
18. Ubeynarayana, C., Goh, Y.: An ensemble approach for classification of accident narratives. In: Computing in Civil Engineering 2017, pp. 409–416 (2017)
19. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480–1489 (2016)

20. Zhang, F., Fleyeh, H., Wang, X., Lu, M.: Construction site accident analysis using text mining and natural language processing techniques. Autom. Constr. **99**, 238–248 (2019)
21. Zhang, J., Zi, L., Hou, Y., Deng, D., Jiang, W., Wang, M.: A C-BiLSTM approach to classify construction accident reports. Appl. Sci. **10**(17), 5754 (2020)
22. Zhong, B., Pan, X., Love, P.E., Ding, L., Fang, W.: Deep learning and network analysis: classifying and visualizing accident narratives in construction. Autom. Constr. **113**, 103089 (2020)