


## Article

# Contextual Representation in NLP to Improve Success in Accident Classification of Mine Safety Narratives

Rambabu Pothina \* and Rajive Ganguli 

Department of Mining Engineering, University of Utah, Salt Lake City, UT 84112-0102, USA;  
rajive.ganguli@utah.edu

\* Correspondence: rambabu.pothina@utah.edu

**Abstract:** Contextual representation has taken center stage in Natural Language Processing (NLP) in the recent past. Models such as Bidirectional Encoder Representations from Transformers (BERT) have found tremendous success in the arena. As a first attempt in the mining industry, in the current work, BERT architecture is adapted in developing the MineBERT model to accomplish the classification of accident narratives from the US Mine Safety and Health Administration (MSHA) data set. In the past multi-year research, several machine learning (ML) methods were used by authors to improve classification success rates in nine significant MSHA accident categories. Out of nine, for four major categories (“Type Groups”) and five “narrow groups”, Random Forests (RF) registered 75% and 42% classification success rates, respectively, on average, while keeping the false positives under 5%. Feature-based innovative NLP methods such as accident-specific expert choice vocabulary (ASECV) and similarity score (SS) methods were developed to improve upon the RF success rates. A combination of all these methods (“Stacked” approach) is able to slightly improve success over RF (71%) to 73.28% for the major category “Caught-in”. Homographs in narratives are identified as the major problem that was preventing further success. Their presence was creating ambiguity problems for classification algorithms. Adaptation of BERT effectively solved the problem. When compared to RF, MineBERT implementation improved success rates among major and narrow groups by 13% and 32%, respectively, while keeping the false positives under 1%, which is very significant. However, BERT implementation in the mining industry, which has unique technical aspects and jargon, brought a set of challenges in terms of preparation of data, selection of hyperparameters, and fine-tuning the model to achieve the best performance, which was met in the current research.



**Citation:** Pothina, R.; Ganguli, R. Contextual Representation in NLP to Improve Success in Accident Classification of Mine Safety Narratives. *Minerals* **2023**, *13*, 770. <https://doi.org/10.3390/min13060770>

Academic Editor: William Skinner

Received: 27 April 2023

Revised: 30 May 2023

Accepted: 1 June 2023

Published: 3 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** mine safety and health; accident narrative classification; machine learning; natural language processing; auto-processing of text; random forests; bidirectional encoding; contextual representation

## 1. Introduction

Text classification has its origins back in the 1960s [1], and machine learning (ML) techniques were successfully applied for text classification in the late 1990s [2,3]. In the recent past, natural language processing (NLP) has many advances in terms of auto-text processing. However, its application is relatively new to the mining industry. Automatic classification of accident descriptions (“narratives”) at mines into their respective categories saves a lot of manhours incurred in documenting and reporting safety-related incidents as part of regulatory compliance. When it comes to pre-training the NLP models for accident classification, US Mine Safety and Health Administration (MSHA) accident database is a valuable resource. In the past multi-year research, several methods, such as RF [4], ASECV, and Stacking [5] were implemented by authors to classify MSHA narratives. Considerable success was achieved, yet certain challenges remain. Mitigating word ambiguity problems from homographs is one of them. In this connection, BERT architecture proved to be very successful and popular [6] in representing words in their context, which effectively

reduces word ambiguity. The aim of the current research is to adapt and implement BERT architecture to the MSHA accident narrative classification problem and to improve upon the success rates achieved by models developed in the previous research. The attempt is the first of its kind in the mining industry.

Datasets used in NLP-based models are at first split into training and test sets. Word embedding is often followed and is simply a representation of words in real-valued vectors that is suitable for use in machine learning (ML) models [7]. The values are arranged in an intuitive fashion; for instance, the closer the words are in vector space, the more similar they are [8]. Contextually meaningful embeddings effectively reduce homonyms or homographs problems. In this manner, each word, even though it is a homograph, can have a unique value in vector space. Embeddings from Language Models (ELMo) [9] and Google's BERT [10] are good examples that adapted context-based word embeddings. The better the representation of the words in their context, the better the fitting of data in the training stage. This will subsequently result in improved prediction performance in the testing stage.

The typical tasks in general for NLP models to accomplish can be to complete a sentence provided by a user [11], answer a question as a chatbot [12,13], or classify narratives into categories [4,5,14]. In the past NLP models, text sequence in a sentence was looked at unidirectionally, either from left to right or from right to left but not in both directions. This has a big disadvantage in finding the right meaning for a word in the given "context" if it is a homograph because the direction chosen can change the context. Bidirectional reading, in the case of BERT, left to right and then right to left, helped reduce the ambiguity by assigning unique values to each word in vector space, even if it is a homograph [10,15]. In addition to BERT, there are other popular NLP architectures in the same arena that adapt contextual representation, such as OpenAI generative pre-training transformer (GPT) [13] and ELMo [9]. The three models are essentially deep learning neural net (NN) architectures that have input, hidden, and output layers. OpenAI GPT uses a left-to-right Transformer while ELMo uses the concatenation of independently trained left-to-right and right-to-left (semi-bidirectional) long short-term memory (LSTM) neural net to generate various NLP "features" such as tokenization, embedding, and so on. [10,16]. Due to this reason, ELMo is called a feature-based model compared to BERT and OpenAI GPT, which are considered fine-tuning-based models when it comes to their implementations on specific tasks. Fine-tuning is a transfer learning process where pre-trained models are tuned in terms of their hyperparameters when applied to specific tasks [17]. More background on the fine-tuning process of BERT is provided in the methodology section.

The challenge, however, for the above models is the availability of large datasets for training. For pre-training BERT, the Toronto BooksCorpus dataset with 800 million words and English Wikipedia with 2500 million words were used [18]. For next-word prediction and generating human-like text, GPT-3 model from OpenAI uses a 2048-token-long context and 175 billion parameters [19]. Masked language modeling (MLM) is another training concept adapted in BERT [10,20] to improve model performance. It is used to predict missing words in a sentence where words are intentionally hidden and are predicted later based on their probability of occurrence in similar contexts. Together with the next sentence prediction (NSP) technique, training set is fitted by BERT to boost model performance on the test set.

Due to its superior contextual representation capabilities, BERT is adapted to many disciplines with specific changes to its architecture and structural features (hyperparameters). Robustly optimized BERT (RoBERTa) is an example where the next sentence prediction of BERT is replaced by a dynamic masking technique where masked tokens change during training epochs. Certain hyperparameters, such as learning rates and batch sizes, are also changed to improve BERT performance [21]. Training BERT on discipline-specific large corpora seems to improve its performance. Examples include SciBERT, which is pre-trained on multi-domain corpora of scientific literature [22], FinBERT that is pre-trained on large financial corpus and fine-tuned for financial sentiment classification of text [23],

and BioBERT, a biomedical text mining model, pre-trained on biomedical domain corpora for biomedical literature [24].

When it comes to specific fields such as construction and the mining industry, the lack of availability of such domain-specific large public databases poses a grand challenge for pre-training. However, when data availability is scarce, or in the case of smaller datasets, pre-trained models such as BERT are helpful in improving the model prediction accuracy [25]. Pre-training avoids the need for training models from scratch for each specific task since such models can be used repeatedly. For instance, short texts from news headlines provide a minimal amount of text to make an opinion out of it. Classification of such news text into true or fake categories becomes a difficult task. In a related research study, BERT is successfully applied to classify short texts from a large public fake news database [26].

In addition to the scarcity of text, the context of words becomes even more challenging when industry-specific vernacular is present. For instance, ‘pin’ is a verb used to denote something that is pinned between two objects; however, a ‘pinner’ is the mining industry slang for a roof bolter, a professional that works in an underground mine. Coming to special or specific industries such as construction and mining, the application of BERT is yet to be explored. One recent application occurred in the construction industry to classify near-miss incidents, and the BERT-based model produced classification accuracy at a rate of 86.9% [27]. In this context, fine-tuning aspect of the BERT architecture seems to be helpful in its adaptation to specific fields, such as the mining industry. However, the aspect has not been adequately or entirely explored by researchers in the mining industry. In this context, authors have adapted the BERT model to the MSHA accident database in the hope of achieving superior narrative classification success rates compared to the approaches tried in their past research.

## 2. Research Methodology

For complete background on the past methodology used, the authors suggest referring to Ganguli et al., 2021 [4] and Pothina et al., 2022 [5]. First, data collection and preparation aspects of model building are presented in Sections 2.1 and 2.2, respectively. To provide proper context for the current research, a few methods used in the past research with key results are presented in the subsequent Sections 2.4 and 2.5, respectively. Section 2.6 is dedicated to the development of MineBERT and its implementation in MSHA narrative classification.

### 2.1. Data Collection: MSHA Accident Database

In order to be consistent and to allow comparison of NLP models in terms of success rates, an MSHA accident narrative dataset collected for the years 2011 through early 2021 is used for the past and current models presented in this paper. There are 81,298 accident narratives in the MSHA dataset, with each narrative having a typical length of five (5) sentences. The dataset is considered “small” when compared to the scale of datasets (large) that are needed for training NLP models such as BERT. A sample (original) narrative from the MSHA dataset can be seen in Table 1. As shown in the table, the narratives are typically converted to their lemmatized forms before being tokenized by the NLP models. For each narrative processed, MSHA personnel manually assigns a category type in the form of a phrase, as shown in Table 1. In this context, it should be noticed that the original narratives and accident types from the MSHA dataset are fed to the NLP models as part of the training set.

**Table 1.** Typical MSHA Narrative and its lemmatized form [5].

MSHA Narrative	Text
Original	“Employee was assisting 3 other miners move Grizzly component in place. While maintaining a vertical position on the component to rehook, the component became unstable and shifted. The employee’s effort to maintain it upright failed and it leaned, pinned his elbow against the rib, bending back and breaking left wrist.”
Lemmatized form	assist 3 miner move grizzly component place maintain vertical position component rehook component become unstable shift’s effort maintain upright fail lean pin elbow rib bend back break left wrist
Accident type	Caught in, under or between a moving and a stationary object (CIMS)

### Selection of Accident Categories for the Study

There are 45 different accident categories in the MSHA database. However, within the scope of the research, nine (9) accident types are only analyzed for RF and BERT models. For SS and ASECV models, certain narrow categories from RF application were of the focus, i.e., “caught-in” and “caught in, under or between a moving and a stationary object (CIMS)” due to the fact they have registered the lowest prediction success rates. Narrow categories have shared some vocabulary with other categories, raising ambiguity problems for algorithms. The major group of accidents that were analyzed in the previous and current research and the MSHA subcategories that fall under each group can be seen in Table 1 from the paper, Ganguli et al., 2021 [4]. The major group of accidents or “Type Groups” are more common, and a large number of accidents fit into those groups. The Type Groups are Caught-in, Fall, Over-Exertion (OE), and Stuck.

Due to the availability of a large percentage of narratives in the Type Groups (major), NLP models, in general, train well. In contrast, due to less availability of narratives from narrow groups and the common vocabulary they share with major groups (raises ambiguity), it is always difficult for models to perform well in such groups. In order to make the models perform well for groups of all ranges, five narrow groups were also chosen as part of nine categories. The following are some abbreviated forms used to identify such groups; over-exertion in lifting objects (OEL), over-exertion in pulling or pushing objects (OEP), fall to the walkway or working surface (FWW), caught in, under, or between a moving and a stationary object (CIMS), and struck by flying object (SFO). The training and test set split ratio among different accident categories for all the NLP models used (past and current) in the research is always 50:50. For detailed numbers on data split for each category, refer to Table 2 of the paper, Ganguli et al., 2021 [4].

### 2.2. Data Preparation for Training and Test Sets

There are 81,298 total narratives in the MSHA dataset used in the research, yielding 40,649 narratives for each of the training and test sets (50:50 split). This is consistent throughout the implementation of the RF, SS, and ASECV models used in the previous research [4,5] and the BERT-based model (MineBERT) developed in this paper. The only exception is that MineBERT uses 10% of the training set narratives for the “validation” purpose (validation set) as part of the original or popular convention. However, the test set and the narratives used in the set are consistently the same among all models since it is the ultimate set that is used in the model performance evaluation. The convention also allows performance comparison of the models in a consistent way. During the training process, typically, models learn from the “training set”, while in the testing stage, the models predict the categories from processing “test set” narratives.

In general, the following steps are typically performed as part of data preparation for the past models and current models presented in the paper. Each narrative is at first converted to lowercase, certain common words that do not add much value to modeling called “stop words” such as in, the, between, employee, EE, etc., along with certain symbols such as “, & /, etc., and spaces are removed. Ultimately, the words are lemmatized or reduced to their root forms; e.g., “pinched” and “pinching” will become “pinch”. In the next step, tokenization and vectorization of words (word embedding) are performed to represent the words in number forms.

### 2.3. Performance Metrics

In order to compare the results across the models, certain performance metrics were used. The “overall success” metric indicates the fraction (%) of narratives in an accident category predicted correctly, either false or true. This can be misleading since the model might have falsely predicted some of the narratives into a certain category called False Positives. Thus, the difference between the number of narratives from “Overall Success” and the number of “False Positives” can provide the number of narratives “Accurately Predicted”. The “Accurately Predicted” and “False Positives” in percentages out of the total number of narratives in a category are two important measures that indicate the performance of a model. These performance measures are adapted throughout past and current research. The aim of the research is to improve “Accurately Predicted” rates to upwards of 90% and minimize the “False Positives” to preferably below 1%. For a detailed example of how the metrics are calculated, refer to the results section of the paper on previous research, Ganguli et al., 2021 [4].

The immediate paragraphs are meant to provide background on past methods (RF, ASECV, and Stacked) followed by BERT implementation methodology.

### 2.4. Previous Research: Random Forest Classifier

Random forest (RF) models are supervised ensemble methods that are popular among the other classification methods due to their robustness and accuracy [28]. The method uses a set of decision trees in order to classify narratives into their respective accident categories. During the “training” process, the RF method learns from the training set and builds decision trees based on each column (“feature”) of the dataset. Then the fitted decision tree model is applied to the narratives in the “test set” to predict their respective accident categories. The method is described more in the works of Ganguli et al., 2021 [4] and Mitchell, 1997 [29]. The success rates and false positives registered by RF implementation can be found in the results section of this paper. The success rate (“% from category accurately predicted”) registered across the board for major or Type Group categories of the MSHA dataset was 75% on average. While Type Group categories registered high success rates, narrow categories were limited to low numbers (25%–59%), which affected the overall success across all categories. The common vocabulary shared by Type Group and narrow categories is the major (ambiguity) problem for the models that detrimentally affected the success rates for narrow groups.

### 2.5. Previous Research: Similarity Score (SS), ASECV, and “Stacked” Approaches

Studies that depend on word frequencies and co-occurrences [30] to predict certain outcomes in NLP are not uncommon. The similarity score (SS) and ASECV methods depend on word frequencies and scoring of certain co-occurring phrases that are key to classification success. A detailed account of the methods is provided in the previous research by Pothina et al., 2022 [5]. Since the two approaches did not improve upon the RF success rates, a “Stacked” approach was devised. Through the Stacked approach, narratives from the test set are tried by RF first and subsequently by ASECV and SS models. By this approach, individual drawbacks of each model can be compensated to improve the overall classification success. When the Stacked approach was implemented for the “Caught-in” accident category, the “% Accurately Predicted” moderately improved from



71% (RF) to 73.28% at a desired level (<1%) of false positive rates. This is encouraging yet not significant enough.

It was identified that all the models from past research suffered from ambiguity problems arising from common words shared among narrative categories. It was also recognized that representing the homographs with unique vector values in their given context can solve the problem. This is the reason behind the selection of BERT architecture for adaptation and implementation in the current research.

#### *2.6. Current Research: Bidirectional Encoder Representations from Transformers (BERT) Approach*

The BERT-based model development involves dividing of MSHA dataset into training, validation or development, and test sets. The validation set is used to evaluate the accuracy of the model and fine-tune the hyperparameters to different versions of the model [31]. Hyperparameters are high-level architectural features of the model. Adjusting (“generalizing”) the hyperparameters for various model scenarios can improve the overall model prediction or classification performance when deployed on an unknown dataset which is generally the test set. Hence, the validation set is key in the development of a successful BERT-based model that works for a wide range of input parameters or patterns. This is the reason the validation set is also called the development set. Due to its comparatively better F1 score (0%–100%), an indicator of model performance, fine-tuning approach with BERT is chosen for the model development in this paper rather than feature-based ELMo [10]. The higher the F1 score, the better the model performance. Due to pre-training, the linguistic patterns are already learned by the BERT-based models before they can be fine-tuned to local or task-specific datasets. In the case of BERT, fine-tuning is achieved by adding one additional outer layer to the existing encoder layers, which are usually frozen. After that, the whole BERT model is trained one more time to achieve the best performance. The training is usually done for 2–3 epochs with sampling batch sizes of 16 to 32.

#### *Development of MineBERT for MSHA Dataset*

Since the BERT model is adapted for the mining industry-specific task in this study, the resulting model is named MineBERT. Certain parameters are specifically chosen to fit the needs of the study, which are presented in Table 2. The unique nomenclature helps distinguish the adapted model from the original and can serve the same purpose for future research and development. Uncasing the text, in general, improves processing speed when the case of the text doesn’t affect model performance. Through past implementations, it is found that the case of the text from the MSHA dataset doesn’t affect the classification performance of the model; hence, the text narratives are uncased before feeding to MineBERT.

There are two major types of BERT models. One is BERT-large that uses 24 encoders, 16 bidirectional self-attention heads, and 340 million parameters. The other is BERT-base that uses 12 encoders, 12 bidirectional self-attention heads, and 110 million parameters [10]. Due to its large size and number of encoders, BERT-large performs slightly better than BERT-base. The computational power required for BERT-large is higher compared to BERT-base. However, their F1 scores on the validation set (96.4% and 96.2%, respectively) and test set (92.8% and 92.4%, respectively) are comparable [10]. Due to this reason, BERT-base-uncased is used for the current research. For MineBERT, 15% of the tokens are masked in the pre-training stage.

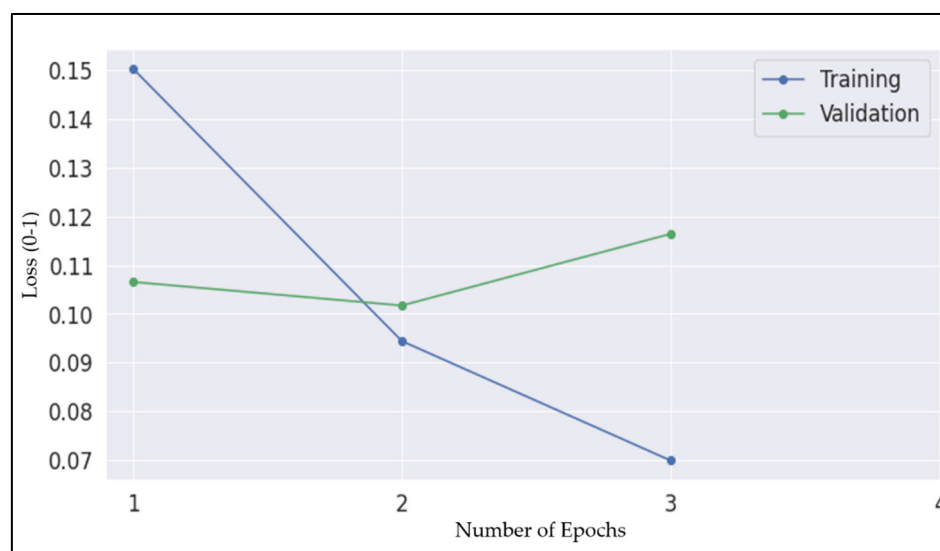
Choosing the right number of epochs and batch size is a challenge, and the model needs to be run on input data many times in order to choose optimum values. Training and validation loss (range: 0 to 1) must also be considered in the context; the minimum the loss, the better the model fitting performance for training data. After many trials, for the purpose of the MSHA dataset, a batch size of 32 was chosen with 2 epochs. Figure 1 illustrates training and validation loss over the number of epochs for the “Caught-in” category. It can be observed that at the epoch number 2 the loss is minimum for both training and

validation. Likewise, for all the other categories, epoch size 2 at batch size 32 provided the best performance.

**Table 2.** Important features adapted for the MineBERT model development.

Parameter	Value/Information
NLP model adapted	BERT-base-uncased
Type of encoder	Bi-directional Transformer
Number of encoders	12
Number of self-attention heads	12
Total parameters	110 million
% of tokens used in masked LM	15%
Type of text processed	Uncased MSHA accident narratives
Major data processing steps	Pre-training followed by fine-tuning
% data split	Training set: 50% (training: 40%, validation: 10%) and Test set: 50%
Training batch size	32
Number of epochs used for fine-tuning	2

It should be recognized that due to the reason BERT architecture has its own tokenizer, full narratives are fed to the model contrary to the RF and other models, which are fed with narratives after processing with NLTK tokenizer.

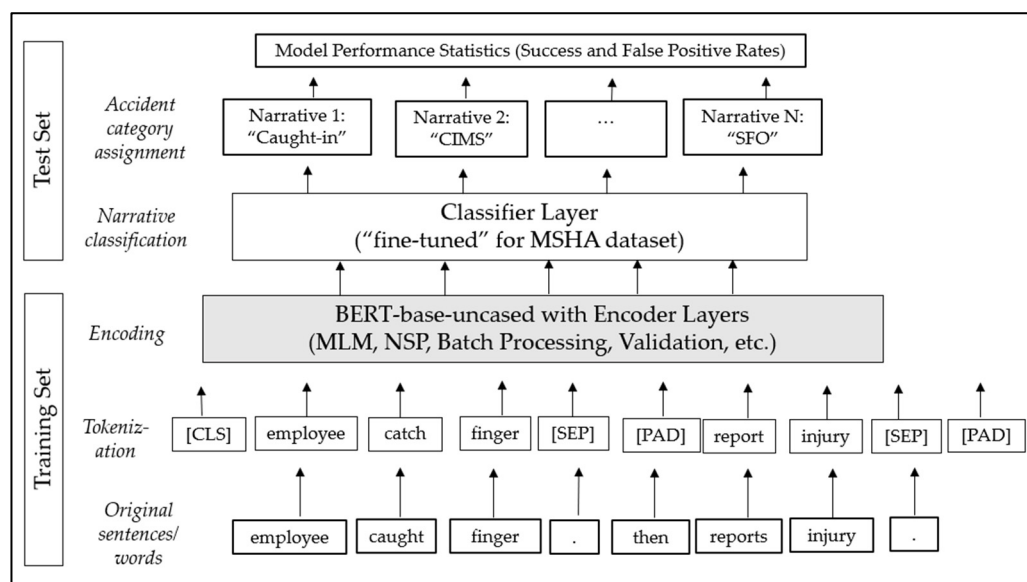


**Figure 1.** MineBERT: Training to Validation loss for the “Caught-in” category.

The following are the major steps involved in the BERT architecture implementation. For the purpose of the paper, BERT-base-uncased is implemented with the PyTorch ML framework. The BERT-base architecture can be seen in Figure 2.

- To allow comparison with past research, the MSHA accident narratives are divided into 50:50 training to testing subsets. In case of MineBERT, however, the training set is randomly split into two, a sub\_training set and a validation subset. The training occurs on the sub\_training set, while the validation subset is used to ensure the generalization of the training. Overall, the split between sub\_training, validation, and test set is a 40:10:50. This split does not affect comparison with the other methods as it is simply an internal procedure of MineBERT training.
- Narratives from the training set are tokenized, and the length of the longest sentence (“maximum length”) in terms of the number of words will be captured. Then, a vector of maximum length is created for the “word embedding” of each sentence.

- Sentence embedding: In the maximum length vector or tensor space, each sentence is identified starting with the code or notation [CLS], and sentences are separated with [SEP] to recognize individual sentences. Empty spaces in the maximum length vector are padded with [PAD] notation.
- In the training process, 15% of the tokens from sentence embedding (sub\_training set) are masked with a notation [MASK] and predicted during the encoding process. The Transformer based encoders in BERT-base perform training and validation processes at the given (optimum) batch and epoch sizes.
- The last layers of MineBERT are used for fine-tuning the whole model to the MSHA dataset from the knowledge gained from the training process.
- The trained model is then applied to test set narratives to get the accident category assignment. A “classifier layer” is used to assign accident categories to the narratives while in the testing stage.
- Performance metrics such as success (“% from category accurately predicted”) and failure (false positive) rates for the test set are then calculated.



**Figure 2.** MineBERT architecture schematic diagram.

### 3. Results

The following are the results from the MineBERT implementation (underlined text) on the MSHA dataset [Table 3] compared to RF results.

**Table 3.** Performance of MineBERT implementation compared to RF approach [4] on the MSHA test set. Bold and Underline: MineBERT.

[illegible]



When compared to the RF model, BERT-base-uncased implementation registered approximately 2% improvement in “Overall success” rates across all the categories. When it comes to “% from Category Accurately Predicted”, the improvement across the board is 23% which is approximately 13% improvement among major or Type Group categories and 32% among narrow categories [Table 4]. It should be noted that the improvements in double-digit percentages are very significant compared to previous methods, which include RF, ASECV, SS, and Stacked approaches. The false positive rates in MineBERT implementation are kept below 1%, which is better compared to previous approaches that ranged between 1% and 5%.

**Table 4.** Performance metrics of RF model in averages and ranges among major and narrow groups compared to MineBERT.

Metrics	Major or Type Groups		Narrow Groups	
	Average	Range	Average	Range
<i>Overall Success:</i>				
RF	93%	(90%–95%)	96%	(95%–98%)
MineBERT	96%	(95%–97%)	97%	(96%–98%)
<i>% from Category Accurately Predicted:</i>				
RF	75%	(71%–81%)	42%	(25%–59%)
MineBERT	88%	(85%–90%)	74%	(64%–83%)
<i>False Positives:</i>				
RF	3%	(1%–5%)	96%	(1%–2%)
MineBERT	1%	(0%–1%)	97%	(0%–1%)

#### 4. Discussion

It is a challenging task to follow several text processing steps in a consistent way to prepare the MSHA accident narratives for comparative analysis. Different ML-based models demand different procedures and code development requirements. The mining industry has special jargon, and the context differs significantly from the other industries. The fine-tuning aspect of the BERT was very useful in this connection. It allows the model hyperparameters to tailor for specific and small sets, such as the MSHA dataset used in the current research. This is one of the key aspects that is exploited in the paper, which contributed to improved classification success rates. In order to allow a fair comparison of success rates among models, performance metrics should also be maintained consistent with previous models. Such procedures were duly incorporated in developing MineBERT architecture. It is difficult, in general, to adapt every metric consistently across the models. However, an identical test set of 50% proportion from the whole dataset is consistently maintained (across all methods compared in this paper) since test set is the ultimate set that evaluates the true performance of the trained model in prediction or classification tasks.

#### 5. Conclusions

In the history of text processing, natural language processing (NLP) proved to be an important and valuable tool. The recent advances in the NLP area have shown that text classification can be automated with improved accuracy when compared to historical approaches. Accident narrative classification is an important step in accident analysis in the mining industry and hence crucial for improving overall worker safety. Authors in the past have implemented an RF-based method to classify MSHA narratives into nine (9) important categories of interest. In addition, original models such as SS, ASECV, and their “Stacked” approaches were developed to improve upon the past classification success rates. Through these past implementations, it was identified that homographs are creating ambiguity problems for NLP algorithms in terms of embedding, thus preventing high success rates. In this context, BERT architecture is identified as an important milestone in the recent past in reducing text ambiguity. In contrast to the past NLP models, it provides

better context for words in a sentence through bi-directional reading and unique encoding, thus reducing ambiguity raised by homographs.

In order to negate the homographs problem and other related challenges, in this paper, BERT architecture (BERT-base-uncased model) is adapted in creating the mining industry-specific model named MineBERT in order to classify MSHA narratives. Adaptation of BERT architecture presents a set of challenges for discipline-specific sets such as the MSHA dataset. In addition, the maintenance of consistent features, data splits, and performance metrics among past and current models to allow fair comparison of success rates is a great challenge that is undertaken and accomplished in this research. The success rates achieved by MineBERT were commendably high when compared to previous models. In comparison to the RF application alone, MineBERT implementation provided approximately 2% improvement in “overall success” rates across all nine (9) accident categories and a 23% improvement for “% from Category Accurately Predicted”, which is approximately 13% increase among major or Type Group categories and 27% among narrow ones. This is a significant milestone in terms of improving accident narrative classification success rates and is one of the first attempts to adapt BERT architecture to the mining industry problems. Future research includes an examination of other linguistic features in addition to customizing and pre-training MineBERT to databases of particularly larger size. Application of the retrained MineBERT to non-MSHA data (private mine partners) is also on the horizon for future consideration.

**Author Contributions:** Conceptualization, R.P. and R.G.; methodology, R.P. and R.G.; software, R.P. and R.G.; validation, R.P. and R.G.; formal analysis, R.P.; investigation, R.P.; resources, R.G.; data curation, R.P. and R.G., writing—original draft preparation, R.P.; writing—review and editing, R.P. and R.G.; visualization, R.P. and R.G.; supervision, R.G.; project administration, R.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sebastiani, F. Machine learning in automated text categorization. *ACM Comput. Surv.* **2002**, *34*, 1–47. [\[CrossRef\]](#)
2. Joachims, T. Text categorization with support vector machines: Learning with many relevant features. In Proceedings of the 10th European Conference on Machine Learning (ECML '98), Chemnitz, Germany, 21–23 April 1998.
3. Dumais, S.T.; Platt, J.; Heckerman, D.; Sahami, M. Inductive learning algorithms and representations for text categorization. In Proceedings of the Seventh International Conference on Information and Knowledge Management (CIKM '98), Bethesda, MD, USA, 2–7 November 1998.
4. Ganguli, R.; Miller, P.; Pothina, R. Effectiveness of natural language processing based machine learning in analyzing incident narratives at a mine. *Minerals* **2021**, *11*, 776. [\[CrossRef\]](#)
5. Pothina, R.; Ganguli, R. The importance of specific phrases in automatically classifying mine accident narratives using natural language processing. *Knowledge* **2022**, *2*, 365–387. [\[CrossRef\]](#)
6. Rogers, A.; Kovaleva, O.; Rumshisky, A. A Primer in BERTology: What We Know about How BERT Works. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 842–866. [\[CrossRef\]](#)
7. Almeida, F.; Xexeo, G. Word Embeddings: A Survey. Available online: <https://arxiv.org/pdf/1901.09069.pdf> (accessed on 21 May 2023).
8. Jurafsky, D.; Martin, D.J. Speech and Language Processing. Available online: <https://web.stanford.edu/jurafsky/slp3/6.pdf> (accessed on 21 May 2023).
9. ELMo. Available online: <https://allenai.org/allennlp/software/elmo> (accessed on 20 May 2023).
10. Devlin, J.; Ming-Wei, C.; Kenton, L.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2022**, arXiv:1810.04805.
11. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. In Proceedings of the Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems (NeurIPS 2020), Virtual, 6–12 December 2020.
12. Mnasri, M. Recent advances in conversational NLP: Towards the standardization of Chatbot building. *arXiv* **2022**, arXiv:1903.09025.
13. ChatGPT. Available online: <https://openai.com/blog/chatgpt/> (accessed on 2 January 2023).

14. Wang, Y.; Sohn, S.; Liu, S.; Shen, F.; Wang, L.; Atkinson, E.J.; Amin, S.; Liu, H. A clinical text classification paradigm using weak supervision and deep representation. *BMC Med. Inform. Decis. Mak.* **2019**, *19*, 1. [CrossRef] [PubMed]
15. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving language understanding with unsupervised learning. Available online: [https://cdn.openai.com/research-covers/language-unsupervised/language\\_understanding\\_paper.pdf](https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf) (accessed on 10 January 2023).
16. Zhen, H.; Xu, S.; Hu, M.; Wang, X.; Qiu, J.; Fu, Y.; Zhao, Y.; Peng, Y.; Wang, C. Recent trends in deep learning based open-domain textual question answering systems. *IEEE Access* **2020**, *8*, 94341–94356.
17. Howard, J.; Ruder, S. Universal Language Model Fine-tuning for Text Classification. Available online: <https://arxiv.org/pdf/1801.06146.pdf> (accessed on 21 May 2023).
18. Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; Fidler, S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *arXiv* **2023**, arXiv:1506.06724.
19. Hegazi, Y.S. Resilience adaptation approach for reducing the negative impact of climate change on coastal heritage sites through machine learning. *Appl. Sci.* **2022**, *12*, 10916. [CrossRef]
20. Wettig, A.; Gao, T.; Zhong, Z.; Chen, D. Should You Mask 15% in Masked Language Modeling? *arXiv* **2022**, arXiv:2202.08005.
21. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. Available online: <https://arxiv.org/abs/1907.11692> (accessed on 22 May 2023).
22. Beltagy, I.; Kyle, L.; Cohan, A. SciBERT: A pretrained language model for scientific text. *arXiv* **2022**, arXiv:1903.10676.
23. Dogu, T.A. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv* **2022**, arXiv:1908.10063.
24. Jinhyuk, L.; Wonjin, Y.; Sungdong, K.; Donghyeon, K.; Sunkyu, K.; Chan, H.S.; Jaewoo, K. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2020**, *36*, 1234–1240.
25. Duan, J.; Hui, Z.; Qian, Z.; Meikang, Q.; Meiqin, L. A study of pre-trained language models in natural language processing. In Proceedings of the 2020 IEEE International Conference on Smart Cloud (SmartCloud), Washington, DC, USA, 6–8 November 2020.
26. Hu, Y.; Ding, J.; Dou, Z.; Chang, H. Short-Text Classification Detector: A Bert-Based Mental Approach. *Comput. Intell. Neurosci.* **2022**, *2022*, 8660828. [CrossRef] [PubMed]
27. Weili, F.; Hanbin, L.; Shuangjie, X.; Peter, E.D.L.; Zhenchuan, L.; Cheng, Y. Automated text classification of near-misses from safety reports: An improved deep learning approach. *Adv. Eng. Inform.* **2020**, *44*, 101060.
28. IBM: What is Random Forest? Available online: <https://www.ibm.com/cloud/learn/random-forest#:~:text=Providesflexibility%3A> (accessed on 15 April 2022).
29. Mitchell, T.M. Machine Learning. In *Machine Learning*; McGraw-Hill: New York, NY, USA, 1997; Volume 45.
30. Morita, K.; Atlam, E.; Fuketra, M.; Tsuda, K.; Oono, M.; Aoe, J. Word classification and hierarchy using co-occurrence word information. *Inf. Process. Manag.* **2004**, *40*, 957–972. [CrossRef]
31. Goot, R.V. We Need to Talk About train-dev-test Splits. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic, 7–11 November 2021; pp. 4485–4494.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.