Injury Report Labeling Using Transformer-Based and Naïve Bayes Model

Yun-Chung Liu, Keon Nartey, Yu Wu

Abstract

Machine-aided injury report classification can facilitate prompt diagnosis of occupational injuries. A dataset of labeled injury reports, or the *real* dataset, was obtained from a competition organized by NASA-Tournament Lab and National Institute for Occupational Safety & Health (NIOSH). Naïve Bayes and the BERT-based model were used for the document classification task of classifying the event type of an injury report. A synthetic dataset was generated based on injury event frequency and token probability given an event. Both the real and the synthetic dataset were split into training, development, and test dataset. Training time and classification accuracy were compared between the two models. The BERT-based model achieved a test accuracy of 86.8 percent and 89.2 percent on the *real* and *synthetic* dataset respectively with model training time of around 15,000 seconds. The Naïve Bayes model achieved a test accuracy of 73.1 percent and 90.8 percent on the *real* and *synthetic* dataset respectively with model training time of less than 20 seconds. The results indicate the Naïve Bayes model can yield a high accuracy rate when the associated assumptions are met with much less computational resources. The high accuracy of the BERT-based model also implies that it can be further optimized for real-world applications.

Introduction

The aim of this project is to leverage Bidirectional Encoder Representations from Transformers (BERT)[1] and Naïve Bayes models to label injury reports and compare model performances. This task is relevant in various domains, including healthcare, workplace safety, and sports, where the prompt identification and categorization of injuries can lead to timely interventions and improved safety measures targeted injury prevention efforts and resource allocation. By automating the categorization of injury descriptions, organizations can streamline incident analysis, prioritize high-risk areas, and implement targeted preventive measures.

The decision to employ Naive Bayes and a Transformer model is rooted in their respective strengths and suitability for the task. Naive Bayes, a probabilistic algorithm, is chosen for its simplicity and efficiency in handling text classification tasks. Its ability to work well with limited data and fast training times makes it a pragmatic choice for initial classification tasks, especially in scenarios where computational resources are constrained. On the other hand, the Transformer model, known for its state-of-the-art performance in natural language processing tasks, is selected for its capacity to capture intricate relationships within text data. The self-attention mechanism of Transformers enables them to discern nuanced patterns and

dependencies, which can be vital in accurately classifying injury reports with diverse language and contextual variations.

The primary aim of this report is to present a comprehensive analysis of the performance of Naive Bayes and Transformer models in the context of text classification for an injury report dataset. By evaluating and comparing the results obtained from these models, the report seeks to provide insights into their respective strengths, limitations, and applicability to real-world scenarios. Additionally, the report aims to offer recommendations for the practical implementation of these models in injury report management systems, emphasizing their potential contributions to enhancing safety protocols and incident response strategies.

Methods

The *Real* Dataset

The dataset used is from a competition organized by NASA-Tournament Lab and National Institute for Occupational Safety & Health (NIOSH). The goal is to automate the processing of data in occupational safety and health (OSH) surveillance systems. NIOSH serves as a dedicated research agency focusing on the study of worker safety and health. Its mission is to ensure safe and healthful working conditions for every individual across the nation while preserving human resources. NIOSH collaborates with partners worldwide. One notable collaboration involves the use of an injury report dataset from a competition organized by NASA-Tournament Lab and NIOSH. This dataset, obtained from Hugging Face, has been preprocessed and includes key information such as text descriptions of injuries, gender, age, and the corresponding event or exposure leading to the injury.

The dataset contains a *text* column of injury report content and an *Event* column with the corresponding label (See table 1 for examples). The classification system with 48 distinct event codes grouped into seven categories. These categories, ordered by precedence, encompass various types of injuries and illnesses, ranging from violence and injuries by persons and animals to overexertion and bodily reactions. The classification aligns with the Occupational Injury and Illness Classification Manual, a standardized system utilized in coding case characteristics for injuries, illnesses, and fatalities in programs like the Survey of Occupational Injuries and Illnesses (SOII) and the Census of Fatal Occupational Injuries (CFOI)[2].

Table 1. Example injury report and label

Text	**Event**
30YOF AT WORK DOING UNSPECIFIED LIFTING AND STRAINED NECK	71 (Overexertion Involving Outside Sources)
31YOM AT WORK USING A NAIL GUN AND SHOT SELF IN THE FINGER WITH A NAIL PW FINGER	62 (Struck by Object or Equipment)
35YOM FELL BACK WHILE FIGHTING A FIRE DX SHOULDER DISLOCATION SUSPECTED ULNAR ARTERY INJ	30 (Fire or explosion, unspecified)
A 28YOF BURNED ARM ON HOT GRILL AT WORK	53 (Exposure to temperature extremes)
67 YOM FELL DOWN STAIRS AND HIT HEAD ON FLOOR AT WORK DX HEAD INJURY	43 (Falls to lower level)

The *Synthetic* Dataset

The *synthetic* dataset is generated based on the assumption of Naïve Bayes model. The injury report in the *real* dataset was tokenized using the BERT tokenizer provided by Hugging Face.

Probabilities of each token appeared in the *real* dataset for each event were calculated. The *same amount* of injury reports for each event was generated for the *synthetic* dataset based on the calculated token probabilities using the *real* dataset. The length of each generated document is uniformly sampled between the maximum and minimum document length of the *real* dataset.

Preprocessing

The following preprocessing steps were done before model training. First, the texts were turned into lowercase and tokenized using a BERT tokenizer, breaking the text into individual tokens, assigning unique IDs, and adding special tokens '[CLS]' and '[SEP]'. These tokens help the model understand the structure and boundaries of sentences.

The *encode_plus()* method provided by the BERT model is employed for efficient tokenization, attention mask generation, and sequence length standardization. This method ensures that each text sequence is padded or truncated to a uniform length.

Additionally, the project determines the maximum sequence length across the dataset to ensure uniformity in input size. The data is then processed through a custom InjuryDataset class, which manages the conversion of text to a model-compatible format.

We also defined a Data Loader Class. Data loaders batch the data for efficient processing in the training, development, and testing phases. The neural network model, EventClassifier, is built on the BERT base, enhanced with dropout layers and a fully connected layer for injury event classification.

Models and Experimental Settings

Pretrained BERT model provided by Hugging Face was used for the injury report classification task. The model was trained on BookCorpus (11,038 unpublished books) and the English Wikipedia leveraging the attention mechanism proposed by Vaswani et al (2017)[2]. The model structure is depicted in detail in the paper by Devlin et al (2018)[1]. The BERT model outputs activation states summarizing information in a document both along the word order in a document and the attention to neighbor tokens within a predefined window size.

The Transformers library provided by Hugging face provides a comprehensive range of pre-trained models, including the BERT model used in this project. BERT's proficiency in understanding contextual language nuances is pivotal for our objective of classifying injury narratives into specific events. We chose the 'bert-base-cased' pre-trained model for our training due to its capability to understand the context and nuances of cased text. This feature is particularly beneficial for processing injury reports where specific terms, acronyms, or entities might be case-sensitive, thereby enhancing the model's precision in classifying and understanding the text.

The baseline model used for performance comparison is Naïve Bayes, which classifies documents based on event probabilities and token probabilities given an event. The Naïve Bayes model is based on the bag-of-words assumption, which does not take word order into account.

Both the *real* and *synthetic* dataset were randomly split into training, development, and test dataset at a 8:1:1 ratio. BERT-based model is finetuned on the training dataset, hyperparameters were optimized using the development set and model performance was evaluated on the test dataset. For the Naïve Bayes model, the training dataset was used for calculating document frequency and token frequency. Model accuracy was evaluated both on the development and test dataset.

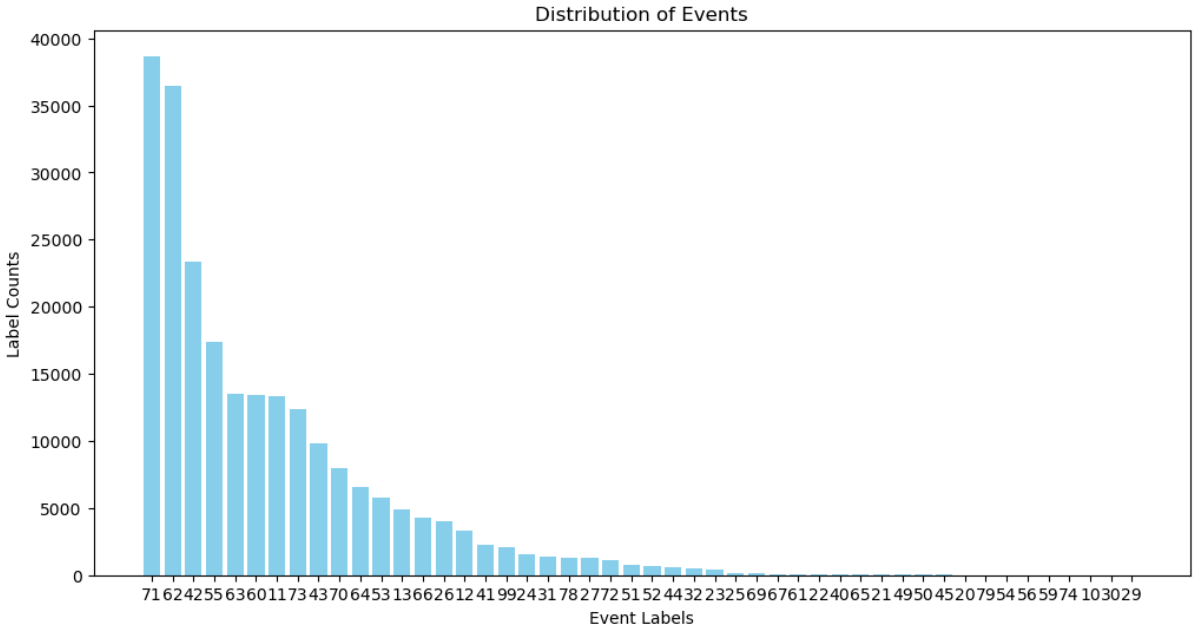
The experiments were conducted using python (version 3.10.12). The pandas (version 2.1.0), numpy (version 1.24.4), matplotlib (version 3.7.2) libraries were used for data preprocessing, analysis, and visualization. Hugging face's Transformer (version 4.35.2) libraries was used for tokenization and for the BERT model. PyTorch (version 2.1.0) was used for the neural network model. Scikit-Learn (version 1.3.1) was used to train the Naïve Bayes model. The BERT model was trained on a T4 GPU provided by Google Colab. The Naive Baye Apple M2 CPU (8-core) on a MacBook Model 14.7.

Results

Datasets

Both the *real* and the *synthetic* dataset contains 222,980 injury reports and 48 different event types. The median count of labels is 724.5 (Interquartile range: 51.3-5100). Both datasets were split into training, development and testing datasets at a ratio of 8:1:1, resulting in 183,856, 22,982, 22,982 rows of dataset respectively. Figure 1 shows the distribution of event count in the *real* dataset, with over 30 thousand reports of event 71 and less than 100 reports for 18 labels.

Figure 1. Distribution of event count in the *real* dataset.



Event 71 - Overexertion Involving Outside Sources (16.83%)

Event 71 indicates injuries resulting from excessive physical effort directed at an external source. This effort may include lifting, pulling, pushing, or carrying objects. The injury can occur from a single episode or repetitive exertions, such as repetitive lifting or pushing and pulling activities. examples; Repetitive lifting of trash cans, files, or luggage; single episodes of lifting furniture or construction materials; pushing and pulling carts or bins.

Event 62 – Struck by Object or Equipment (15.85%)

This encompasses injuries caused by forcible contact or impact between the person and the source of injury. This occurs when the motion producing the contact is primarily from the source of injury. Examples include being struck or run over by vehicles, caught between vehicles, or struck by swinging or slipping objects and finally injuries resulting from separating materials.

Less Frequent Events

Event 29 and 30 both accounted for about 0.0008% - representing injuries arising from transportation incidents and fires or explosions with unspecified details.

Model Parameters

The pretrained uncased BERT model was used for the injury report classification task. The length of the hidden state was 786. A dropout rate of 0.3 was applied to the output state and a fully connected layer was added to output classification results. Each injury report was padded to the maximum document length of 150 tokens. Batch size was set at 16, with a learning rate of 0.00002 for model training. The learning rate was set to decrease linearly during the training stage, across training epochs and. Cross entropy was used as the loss function and Adam was used for model optimization. The BERT model was trained for 10 epochs. For the Naïve Bayes model, the smoothing parameter of alpha was set to be 1.

Model Performance

Table 2 summarized the performances of both models on both the *real* and *synthetic* dataset. The BERT model achieved an accuracy rate of 86.79% on the test dataset and 86.99% development dataset accuracy on *real* data, slightly lower than the accuracy rate on the *synthetic* dataset (89.22% on the test set and 89.13% on the development set). Naive Bayes achieved an accuracy of 73.14% on the test set and 73.13% development set of *real* injury reports. On the *synthetic* dataset, the accuracy rate was 90.77% for the test set and 90.87% for the development set.

Table 2. Model performance on the injury report classification task

	BERT	**Naive Bayes**
Real Dataset		
Test Accuracy	86.79%	73.14%
Development Accuracy	86.99%	73.13%
Train Accuracy	95.83%	75.26%
Synthetic Dataset		
Test Accuracy	89.22%	90.77%
Development Accuracy	89.13%	90.87%
Train Accuracy	99.47%	91.64%

Training Time

Table 3 summarized the wall time spent for model training on both the *real* and *synthetic* dataset. The average training time per epoch for the BERT model is 1477.7 seconds on the *real* dataset and 1483.5 on the *synthetic* dataset. The training time for the Naive Bayes model was 19.06 seconds on *real* data and 11.92 seconds on *synthetic* data. The BERT model was trained for 10 epochs. Thus, the total training time for the BERT model was above 1,200 times longer than the Naïve Bayes.

Table 2. Training time comparison

	Transformer	**Naive Bayes**
Real Dataset	1477.7 seconds/epoch	19.06 seconds
Synthetic Dataset	1483.5 seconds/epoch	11.92 seconds

Discussion

The projects demonstrated the model performance and the time required to train a BERT-based model and a Naïve Bayes model on labeling *real* and *synthetic* injury reports. The BERT-based model achieved a high accuracy rate (86%-90%, on development and test dataset) on both the *real* and *synthetic* data. For the Naïve Bayes model, the accuracy rate was lower on the *real* dataset (83% on development and test data set) and higher on the *synthetic* dataset (90%-91% on development and test dataset). The training time for the BERT-based model, even using GPU), is over 1,000 times longer than Naïve Bayes. The implication is that when the assumptions of the models are met, generative models, such Naïve Bayes, can achieve comparable prediction accuracy as a BERT-based model with much less computational resources.

The performance difference for the Naïve Bayes model on *synthetic* documents and *real* documents reflects how the prediction accuracy can drop when the assumption of the model does not hold. The Naïve Bayes model classifies documents based on the relative frequency of each label (or event) and the probability of tokens given a label, assuming that the frequency of tokens are independent of one another, which is not true in *real* injury reports. This can partly explain the lower performance of the Naïve Bayes model on the *real* dataset.. The *synthetic* dataset was generated using the document frequency and token probability given an event, which explains why the Naïve Bayes model is performing well on the synthetic dataset. In addition, since injury reports are relatively short (20-60 tokens), the generated dataset could actually contain only more frequent tokens given each event, which could potentially explain the increase of prediction accuracy on *synthetic* dataset.

The BERT-based model is performing well on classifying both the *real* and *synthetic* injury reports because it can capture complex relationships and context within text. For the *real* dataset, the context and word order is important for classifying the event, which explains why the

transformed-based model outperformed the Naïve Bayes model. For the *synthetic* dataset, the BERT-based model can learn to ignore word order and context and assign weights to key words related to the event type (label), which is why it can achieve comparable prediction accuracy as the Naïve Bayes model.

The difference in training time is due to the nature of the BERT-based model and the Naïve Bayes model. The BERT-based model has a larger amount of weights to train (e.g. attention to tokens, hidden state) and optimizing these parameters using gradient descent takes a lot of computational resources. On the other hand, training the Naïve Bayes model is calculating event frequency and token probability given an event using any training dataset, which is less computationally demanding.

Limitations

The uneven distribution of events can have a negative impact on prediction accuracy, especially on the less frequent events. In the dataset used, the most common event label represents approximately 16.8 percent of the total number of documents. The least common labels represent less than 0.01 percent of the total number of documents, making model training difficult for both the Naïve Bayes and the transformed-based deep learning model.

Due to computational resource limitation, the amount of training epochs was limited and hyperparameter tuning was not conducted thoroughly. With more computational resources invested, the accuracy rate of the BERT-based model could increase even more.

Future Considerations

To enhance model performance and investigate the feasibility of real-world application, we aim to complete the following:

1. Misclassification analysis: Investigating the prediction accuracy for each label might inform feature engineering effort. Also, indications for application in real-world application can be found during the process. For example, false negatives can be a serious issue for certain labels while false positives can be more troublesome for another event.
2. Feature engineering: Experimenting with feature engineering techniques, such as different text representations or additional relevant features, could enhance model performance.
3. Data Collection: Collecting more labeled injury reports, especially the less common ones, will benefit model training and performance.
4. Hyperparameter tuning: Further hyperparameter tuning (e.g. number of training epochs, learning rate, batch size) could improve model performance.

Conclusion and Implications

In conclusion, the application of text classification techniques to the injury report dataset represents a pivotal advancement in Machine Learning and Natural Language Processing. This task is not merely a technicality but a cornerstone for bolstering safety protocols and incident management within healthcare. The accuracy and precision achieved through classification play a critical role in enhancing customer satisfaction, particularly in healthcare services where precise diagnosis is paramount. Beyond the immediate implications for patient well-being, the financial repercussions are substantial. Accurate documentation and classification translate to reduced chances of denied insurance claims, offering patients a financial reprieve. This is especially significant in cases where injuries may be occupational, rather than intentional, underscoring the broader societal impact of robust text classification in healthcare.

Furthermore, the advantages extend to the operational efficiency of healthcare workers. Automation, coupled with proper documentation and classification, minimizes the time spent on manual data input and inter-departmental record transfers. The resultant time savings can be transformative, allowing for swift and life-saving interventions. This not only streamlines healthcare processes but also positions healthcare professionals to respond more effectively to serious injuries. By contributing valuable insights, we aim to empower decision-makers and practitioners to implement and optimize text classification solutions for injury reports. Such measures not only promise improved resource allocation within hospital systems but also pave the way for continuous enhancements, ultimately working towards a more efficient, responsive, and impactful healthcare ecosystem.

References

1. Devlin Jacob, Chang Ming-Wei, Lee Kenton, and Toutanova Kristina. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186.
2. CDC. (n.d.). CDC Occupational Injury and Illness Classification System (OIICS). Retrieved from https://wwwn.cdc.gov/wisards/oiics/Trees/MultiTree.aspx?TreeType=Event
3. Vaswani Ashish, Shazeer Noam, Parmar Niki, Uszkoreit Jakob, Jones Llion, Gomez Aidan N., Kaiser Łukasz, and Polosukhin Illia. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. 5998–6008.