Disaster Tweet Classification

Kevyn Anguiera Irizarry, Mohit Kankanala, Param Bhavsar

Abstract

This paper presents a study on the classification of disaster-related tweets using the Kaggle "Natural Language Processing with Disaster Tweets" dataset. We address the challenge of correctly classifying disaster tweets if they are about a disaster or not. Our approach includes a data preprocessing, where misclassified tweets are corrected, and irrelevant symbols are removed. We employ sentence transformers for semantic encoding and implement K-Means clustering to refine our labels categorization. The fine-tuning of a pre-trained BERT-base model with our newly generated labels demonstrates a significant improvement in classification accuracy, highlighting effectiveness of our proposed labelling method.

1 Introduction

In the digital age, social media platforms serve as hubs for real-time information, often becoming crucial channels of communication during emergencies. Yet amidst the wealth of user-generated content, distinguishing between posts about real disasters and mundane events poses a pressing challenge. User-generated content ranges across all topics and commonly disaster-related words may be used as hyperbole, metaphors, or in alternate contexts. It is necessary to understand the semantic or contextual purpose of disaster words in social media posts to properly identify disasters from non-disasters. The timely identification of real disasters is crucial for the prompt deployment of aid and crisis management.

In recent years, the emergence of large language frameworks and large language models (LLMs) like Bert , GPT, and their successors have revolutionized natural language processing (Devlin et al., 2019; Lewis et al., 2020). These LLMs possess the capacity to comprehend nuanced contextual cues within contextual data, making them invaluable in text classification. Taking advantage of their high task generalizability, LLMs can be prompted to classify social media posts, specifically Twitter posts or tweets, between disaster and non-disaster tweets. For example, a zero-shot learning classifier can make a prediction on unseen classes during its training phase. A zero-shot learning text classifier would make use of its learned semantic relations during training to classify tweets between disasters and non-disasters. Alternatively, existing LLMs can be fine-tuned using a small number of samples to teach them new semantic relations and optimize their performance at a specific task. In this paper, we will fine-tune Bert on a disaster tweet database to create our classifier.

2 Dataset

The dataset is one of the critical components of our Disaster Tweet Classification project. We imported the dataset from Kaggle competition titled "Natural Language Processing with Disaster Tweets," it comprised approximately 7,000 tweets, each labeled as either 1 or 0, where 1 is for disaster and 0 is non-disaster (Addison Howard and Guo, 2019).

Although we got our dataset from kaggle, which is regarded as highly reliable, we did find some inconsistencies in the data. We found some tweets in Table 2 labelled as disaster when in actuality was about non-disaster and vice versa shown in Table 1. The tweets had non alphabetic characters that are symbols like "", "#", and links, which may not add to the sentiment of the sentence for our purposes.

To address these challenges, we adopted a multi-faceted approach. We first sought out the find most of the falsely classified tweets and change the labels to correctly classify the tweets. We then extracted all the tweets that had mentions-"" and hashtags-"#" and removed them.

After cleaning all the tweets, we then split the Kaggle data into train and test data with the distribution 70% and 30%.

keyword	location	text	target
blood	Cario, Egypt	people with a #tattoo out there.. Are u allowed to donate blood and receive blood as well or not?	1
blizzard		I call it a little bit of your blizzard?	1

Table 1: Tweets are falsely classified as disaster

keyword	location	text	target
drowning	LONDON	LONDON IS DROWNING AND IIII LIVE BY THE RIVEEEEEER	0
bioterror		FedEx will no longer transport bioterror pathogens in wake of anthrax lab mishaps	0

Table 2: Tweets are falsely classified as non-disaster

3 Methodology

Our main goal is to sort tweets into two groups: real disaster tweets and non-disaster tweets. To achieve this, we will utilize the "Natural Language Processing with Disaster Tweets" Kaggle database for training and testing (Addison Howard and Guo, 2019). Moreover, we will benchmark the performance of three models in classifying the data:

1. Zero-Shot Learning Classifier (baseline)
2. Fine-tuned LLM on Kaggle Labels (experimental 1)
3. Fine-tuned LLM on Cluster Labels (experimental 2)

3.1 Initial Exploration

Before any work, we must analyze our dataset to ensure its adequacy for our task. We will use the Kaggle "Natural Language Processing with Disaster Tweets" dataset (Addison Howard and Guo, 2019). This dataset contains a sequence of 7614 entries of tweets classified between disaster and non-disaster. Each entry includes text, keywords, location, and classification. We first performed a word frequency analysis to ensure significant differences between the classes. Then we analyzed the data labeling to ensure the dataset was accurately labeled. The analysis found that a significant portion of the dataset was mislabeled, which would degrade model quality. As such, we will develop a new set of alternative labels based on unsupervised clustering.

3.2 Preprocessing

To generate our set of new labels we must first create representational sentence embeddings for our dataset. These embeddings will encode our sentences based on semantic interpretations. As such, noisy and irrelevant data may provide inaccurate embeddings. We performed tests removing combinations of twitter links, account mentions, and hashtags. We concluded that removing twitter links, account mentions, and hashtag symbols but not the words provided the best performance.

3.3 Sentence Encoding

Once denoised, we must use a sentence transformer to encode the text from the dataset into sentence embeddings. These embeddings are crucial for clustering, as traditional clustering algorithms require numeric values rather than text. We tested the performance of two of the top sentence transformers: Universal Sentence Transformer (USE) and all-mpnet-base-v2, as seen in figure 3 (Cer et al., 2018; Song et al., 2020). Sentence encoders transform the text into a 768-dimensional array. We decided to use all-mpnet-v2 as it provided the best performance.

3.4 Dimensionality Reduction

Our sentence encoding resulted in a high dimensional 768-dimension array. As higher dimensional inputs require a large input to train and our dataset was relatively small, approximately 7k entries, we contemplated dimensionality reduction algorithms like Principal Component Analysis (PCA). PCA reduces the dimension by conserving the largest dimensions with the highest correlation (Maćkiewicz and Ratajczak, 1993). However, we saw that dimensionality reduction had a negative impact on our performance as seen in figure 3. We concluded that PCA prioritized similarity and variance representation in data which may not correlate properly

with key class attributes.

3.5 Clustering

To generate the new labels for our dataset we must first create unsupervised clusters. We attempted two different strategies: a distance-based approach through K-Means and a density-based approach through DBSCAN (Hartigan and Wong, 1979; Ester et al., 1996). We found that on average K-Means matched 77.58% of the original labeling. Upon manual inspection of mismatched cases, we concluded that K-Means more accurately handled the false negatives and false positives found in the original dataset. Alternatively, when we attempted to run DBSCAN we found the data could not be properly separated based on density, approximately a third of the data was considered noise and could not be clustered (Ester et al., 1996). Due to the adequate performance metrics, we chose to cluster by K-Means.

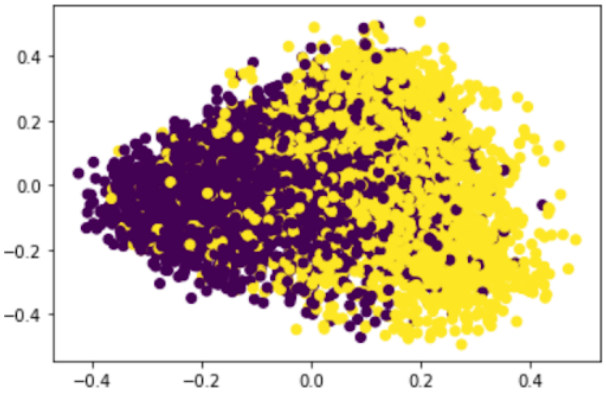

Figure 1: 2D visualization of labels before clustering.

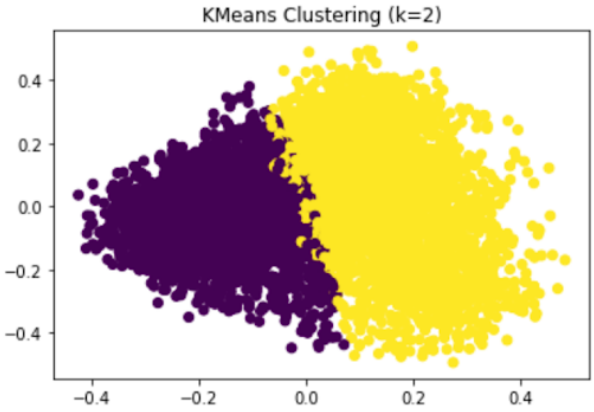

Figure 2: 2D visualization of labels after clustering.

3.6 Labelling

Once we develop the clusters, we must identify each cluster as either the disaster cluster or the non-disaster cluster. As unsupervised clustering there is no pre-encoded label. Instead, each cluster will be assigned the predominant label from the original dataset. We see in Figures 2 and 3 the PCA visualizations for the original labels and the new labels, respectively. We see how the new labels eliminate most of the noise in the dataset creating a clear boundary between labels.

3.7 Fine Tuning

To optimize our model's performance, we used the BERT (Bidirectional Encoder Representations from Transformers) architecture by utilizing the pre-trained BERT-base model with 110 million parameters (Devlin et al., 2019). This choice provided a solid foundation for natural language understanding tasks.

To align the model with our domain and task, we undertook a fine-tuning process that involved both the original labels and newly generated labels from our proposed methodology. The original labels formed the baseline, while the incorporation of newly generated labels enhanced the model's understanding of domain-specific nuances.

Our fine-tuning procedure spanned three epochs, strategically balancing computational efficiency and model convergence. Each epoch constituted a comprehensive pass through the entire training dataset, and this limited epoch count was deliberate to prevent overfitting while enabling the model to adjust to task intricacies. We adjusted the model's parameters throughout fine-tuning to optimize its ability to discern patterns within the dataset. Consequently, the fine-tuned model showcased improved performance, highlighting its adaptability to the refined labels and enriched representations generated by our proposed methodology. This fine-tuning process played a pivotal role in our methodology, bridging the general language understanding capabilities of the pre-trained BERT-base model and the specific nuances introduced by our domain and task. The resultant model exhibited heightened proficiency in capturing data intricacies, ultimately enhancing performance on the targeted classification task.

Overall, our methodology encompassed a blend of data preprocessing, advanced semantic encoding, strategic clustering, and careful fine-tuning of a lan-

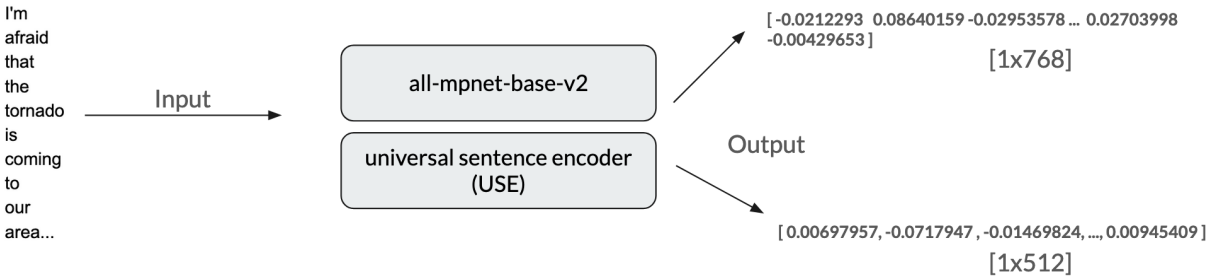

Figure 3: Sentence Encoding Work Flow.

Model	Accuracy
use + pca(95%) + K-means	54.32%
all-mpnet-base-v2 + PCA(95%) + K-means	56.59%
use + K-means	72.73%
all-mpnet-base-v2 + K-means	**77.58%**

Table 3: The accuracy assessment of various variants of the K-Means clustering algorithm employed to create new labels for a given dataset.

guage model, all aimed at improving the accuracy and reliability of disaster tweet classification.

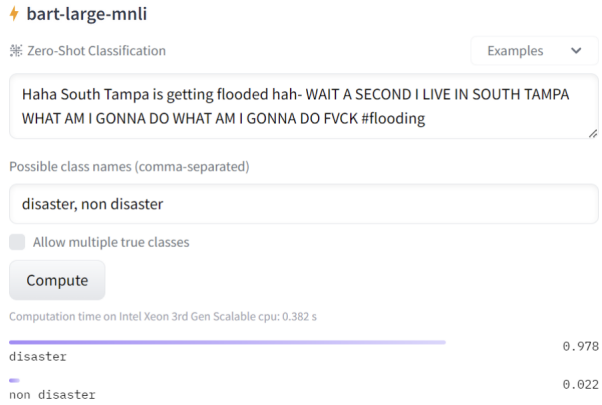

Figure 4: Zero-shot classification accuracy.

4 Conclusion

5 Results

Our proposed method for generating dataset labels has shown highly promising outcomes in the context of the Disaster Tweet Classification task. This underscores the effectiveness of our devised methodology. Through the fine-tuning of the BERT-base Language Model, incorporating newly derived labels from our clustering-based label generation method has led to a significant enhancement in classification accuracy. Examination of the results presented in Table. 4 reveals an approximate 8% increase in accuracy when utilizing these updated labels as opposed to the original ones. This result validates our strategy, which integrates sophisticated clustering techniques into the model training process, and emphasizes the critical role of precise and nuanced labeling in advancing the outcomes of machine learning applications.

The improved capability of our model to accurately categorize tweets related to real disasters underscores the practical utility of our approach. This has profound implications, particularly in enhancing automated disaster response systems within the realm of social media. The success of this research project serves as evidence of the efficacy of combining advanced data preprocessing, clustering methodologies, and model fine-tuning techniques within the domain of natural language processing. Furthermore, it highlights the potential for leveraging Language Models like BERT to strengthen underlying language understanding, making the model more versatile and adept at handling unforeseen variations in tweet content.

In conclusion, our research has effectively addressed the hypothesis that zero-shot performance is suboptimal for our tweet disaster classification task. We have illuminated the challenges posed by noisy labels in the dataset, prompting the need for a novel approach to label generation. Leveraging unsupervised methods, specifically a clustering technique based on embedding values, allowed us to generate new labels that are more accurate and meaningful in capturing the nuances of tweet content. Our findings have challenged the assumption that top-performing generalized language models

Fine-tuned Model	Accuracy
BERT-base (old labels)	82.79%
BERT-base (new labels)	**90.49%**

Table 4: Evaluation of BERT-Based Model Accuracy on Old and Newly Generated Tweet Labels

inherently excel in the specific task of tweet classification. The suboptimal zero-shot performance demonstrates the importance of tailored solutions for domain-specific applications.

Furthermore, fine-tuning an LLM like Bert using these refined labels has proven to be a pivotal step in enhancing performance. The notable improvement of approximately 8% attests to the efficacy of our approach, highlighting the significance of incorporating unsupervised methods for label generation and subsequent fine-tuning to achieve superior results in tweet disaster classification tasks. This research not only contributes to understanding the limitations of generalized language models but also provides a practical solution to enhance their performance in specific domains through careful label curation and model refinement.

Acknowledgements

Gratitude permeates our acknowledgments as we extend heartfelt thanks to Professor Chris Brew, whose invaluable guidance and insightful feedback shaped this research profoundly. The OSU Supercomputing Center (OSC) deserves our appreciation for providing essential computational resources. The internet's vast trove of information and collaborative platforms significantly enriched our study. Our trusty computers deserve recognition for their unwavering reliability, ensuring uninterrupted progress through extensive computations. Lastly, we extend our thanks to Kaggle for offering the pivotal Disaster Tweet dataset, serving as the bedrock for our explorations in disaster tweet classification.

References

Phil Culliton Addison Howard, devrishi and Yufeng Guo. 2019. Natural language processing with disaster tweets.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press.

J. A. Hartigan and M. A. Wong. 1979. A k-means clustering algorithm. *JSTOR: Applied Statistics*, 28(1):100–108.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Andrzej Maćkiewicz and Waldemar Ratajczak. 1993. Principal components analysis (pca). *Computers Geosciences*, 19(3):303–342.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.