Mini-Project 3: Classification of Textual Data

Yijie Zhang
McGill University
yj.zhang@mail.mcgill.ca

Tian Bai
McGill University
tian.bai3@mail.mcgill.ca

Yiping Liu
McGill University
yiping.liu@mail.mcgill.ca

Abstract

In this project, we implement the Naive Bayes model and the BERT-based model for emotion classification, and we test the performance of different models on the Huggingface emotion dataset. The primary discovery of this project is that the BERT-based model significantly outperforms the Naive Bayes model due to their capability of capturing correlations between features. However, the complexity and cost of the BERT-based model exceed that of the Naive Bayes model. In general, we should wisely choose our models so that they align with both our requirements and constraints.

1 Introduction

The text classification problem dates back to the mid-20th century. It holds significant value in various domains and applications, ranging from information retrieval to spam filtering, intelligent customer service, and social media analysis. As technology advances, text classification tasks will continue to evolve, presenting new challenges and opportunities (1).

The Naive Bayes method has been extensively studied since the 1950s and was introduced to text information retrieval in the early 1960s. Until today, it remains a popular benchmark method for text classification, which is characterized by word frequency to determine the category of a document or other issues (such as spam, legality, sports, politics, etc.) (2). It is a class of probabilistic models based on Bayes' theorem. It simplifies the dependency relationships between features given a class, making calculations more straightforward. The advantages of Naive Bayes include simplicity, efficiency, and ease of implementation, particularly performing well in cases with small datasets. However, due to its naive assumption, it may not perform well when features have strong interdependencies.

BERT (4) was created and released in 2018 by Jacob Devlin and his colleagues at Google. BERT originates from pre-training context representations, including semi-supervised sequence learning, generative pre-training, ELMo, and ULMFit (3). Unlike previous models, BERT is a deep bidirectional, unsupervised representation of language that is pre-trained using only a plain text corpus. No context models such as word2vec or GloVe generate a single-word embedding representation for each word in the vocabulary, where BERT takes into account the context of each occurrence of a given word. It is a pre-trained language representation model based on the Transformer model. It uses attention mechanisms to capture long-range dependencies and includes two stages of pretraining and fine-tuning. During the pre-training, the model learns general language representations through a large-scale language modeling task and then fine-tuned for specific tasks.

The primary project of this project is a) implement the Naive Bayes model and the BERT-based model for emotion classification and test their performance; b) examine the attention matrix between the words and class tokens for some of the correctly and incorrectly predicted documents.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

In the experiments, we observed that the BERT-based model significantly outperforms the Naive Bayes model. Pretraining helps with capturing features of complex text and benefits the prediction task. However, the computational complexity and cost of the BERT model also increase. To balance the cost and accuracy, we should analyze the dataset carefully and choose a more suitable method based on specific tasks and features of the dataset.

2 System Models

2.1 Implementation of Naive Bayes Model

The Naive Bayes model is particularly known for its simplicity, efficiency, and effectiveness, especially in tasks like text classification and spam detection. In our project, we implemented two versions of the Naive Bayes Model, respectively based on multinomial distribution and bernoulli distribution, for the emotion classification task. Our approach integrates a categorical prior distribution with a multinomial/bernoulli likelihood to account for word frequencies. It also employs Laplace smoothing with pseudocount $\alpha = 1$ to handle the zero-frequency problem, enhancing the model's robustness and accuracy in predicting text categories.

2.2 Implementation of BERT Model

In this project, we leverage the BERT model for emotion classification, processing text data through the HuggingFace BertTokenizer and BertForSequenceClassification. Our approach tokenizes and prepares the datasets into TensorDatasets and adapts the model to classify text into six emotional categories, further enhancing its capability by utilizing attention outputs for deep insights into the model's decision-making process. We trained our model based on the 'bert-base-uncased model' in the HuggingFace library, and compared its performance with the model 'bhadresh-savani/bert-base-uncased-emotion'.

3 Experiments and conclusion

3.1 Dataset Overview

The emotion dataset is a dataset of English Twitter messages with six basic emotions such as anger, fear, and sadness. The data fields contain text, which is a string feature and a classification label for the emotions. The split version is 20,000 examples split into train, validation, and split datasets; the unsplit version is 416,809 examples in a single train split dataset. In this project, we only utilize the split version and build our own features.

3.2 Data Preprocessing

To verify the feasibility of the experiments, we preprocessed the datasets. For the Emotion Classification dataset, we extracted some samples and visualized them with their labels, to better understand the structure of the dataset, shown in Figure 1. As the input of the Naive Bayes model, we utilized the *CountVectorizer* function to transfer the original text and words into feature vectors. Before feeding into the model, we changed all the feature vectors and labels into array format since our model relies on *numpy* package, for which array fits better. For the BERT Model, we adapted the *TensorDataset* class from the Pytorch library since padding and masks are introduced into the model. As with other deep learning models, we used the standard Pytorch Dataloader to split the data into batches and iterate them in the training loop.

```
16000
i didnt feel humiliated
i can go from feeling so hopeless to so damned hopeful just from being around someone who cares and is awake
im grabbing a minute to post i feel greedy wrong
i am ever feeling nostalgic about the fireplace i will know that it is still on the property
i am feeling grouchy
[0, 0, 3, 2, 3]
```

Figure 1: Preprocess of Emotion Classification Dataset.

3.3 Experimental Setting

For the naive Bayes method, we preprocess the dataset by turning the unstructured test data into numerical features using count vectorizer. In the multinomial setting, the count of each possible word is considered as a feature, while in the bernoulli setting, the occurence of each possible word is considered. For BERT, we tokenize the input text and convert the tokens into numerical features with

the transformers package. Then we pack the data into a dataset. The validation accuracy curve of our BERT model is shown in Figure 2.

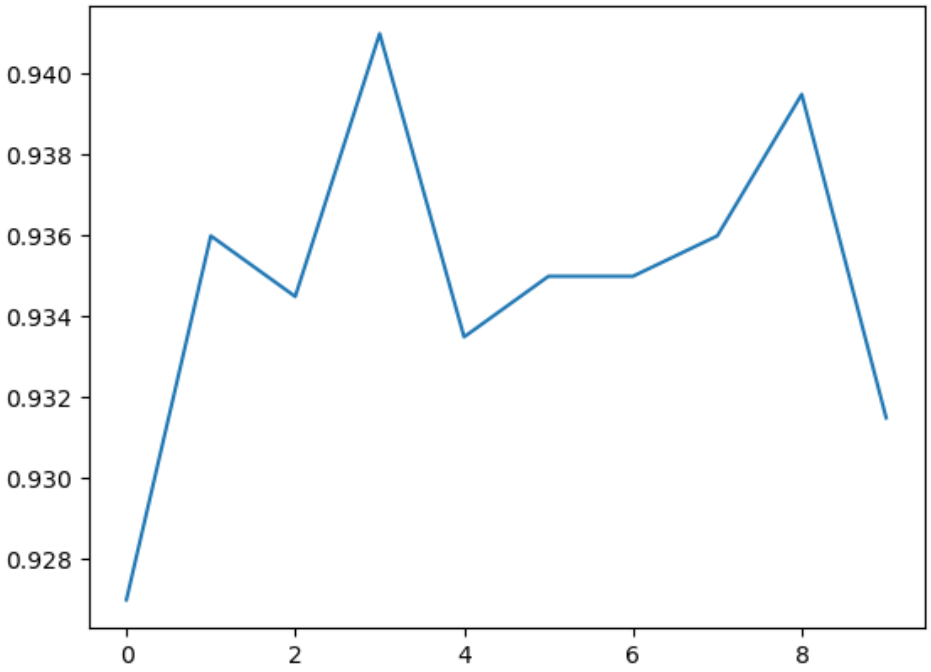

Figure 2: Validation accuracy curve in the training process of our BERT model.

3.4 Performance of the Naive Bayes and BERT

The performance of the the two Naive Bayes models and BERT-based models on the Emotion classification task are shown in Table 1. For the two Naive Bayes models, the multinomial version significantly outperforms the Bernoulli version. This could be explained by the fact that in Bernoulli Naive Bayes, the occurrence of each word is considered as distinct feature, while in Multinomial Naive Bayes the count of words correlates with each other (as the MLE for $\Pr(X_i = x_i|Y = c) =$ portion of x_i in all words with label $Y = c$, so the frequency of words is what matters). Thus the Bernoulli Naive Bayes is less expressive in representing the correlation of words in a sentence, especially when the sentence is long. (7; 8)

Our BERT model has a similar performance compared to that of the pretrained BERT model. In general, BERT-based models perform better than Naive Bayes models. This could be caused by several reasons: 1) Representational Capacity: The Naive Bayes is a simple model based on probabilistic statistics, assuming independence among individual features (vocabulary). It often performs well in simple tasks like text classification but cannot capture complex relationships between words. While BERT uses deep neural networks, especially Transformer models, these models possess more powerful representation learning capabilities (5). 2) Data Scale and Diversity: When faced with large-scale and diverse datasets, the simplistic model structure of Naive Bayes may fail to generalize effectively. By contrast, when trained on large-scale datasets, BERT can better capture language complexity and adapt to different domains and data distributions through fine-tuning. 3) Task Complexity: Naive Bayes deals with relatively simple text classification tasks well, especially when the context of the text is straightforward, and relationships between words are relatively independent. For BERT-based models, because of their powerful representation learning capabilities, they are suitable for more suitable for complex tasks.

Table 1: Performance comparison of 4 models.Attention matrix of correctly predicted documents.

Method	Bernoulli Naive Bayes	Multinomial Naive Bayes	BERT	Pretrained BERT
Accuracy	0.415	0.795	0.92565	0.9265

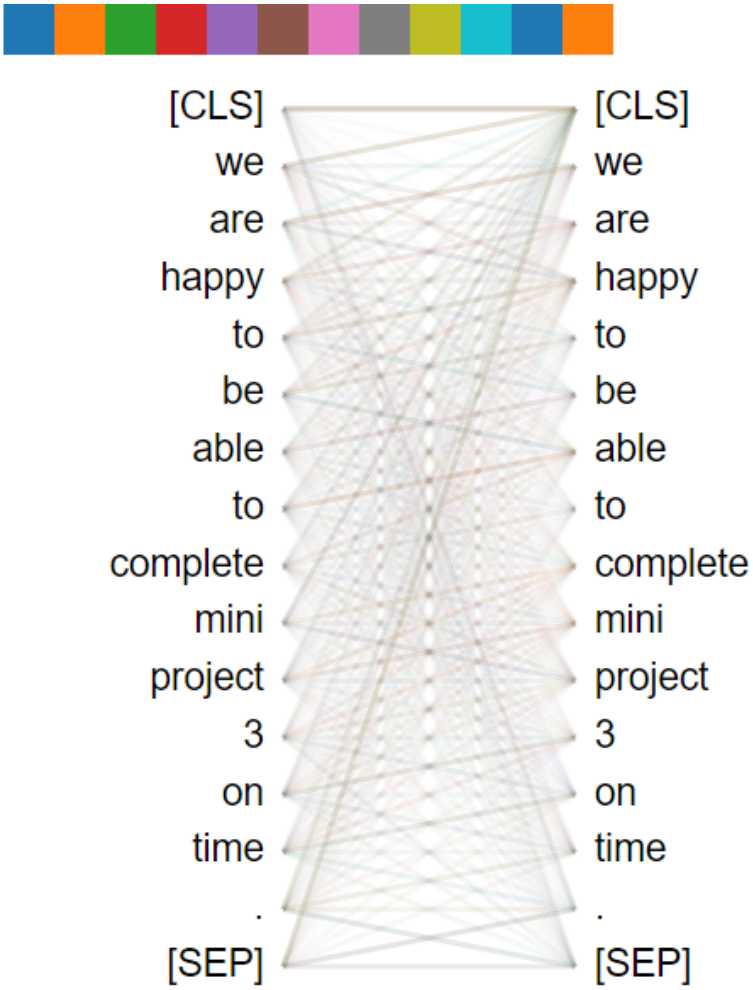

Figure 3: Attention matrix of correctly predicted documents.

3.5 Visualization of the attention matrix

The attention matrix between the words and class tokens is shown in Fig. 3 and Fig. 4. From the results, we can find that pretraining on an external corpus does help emotion prediction tasks.

There are several aspects that can lead to this: 1) contextual understanding: Emotions are always context-dependent, and pretraining allows the model to capture the relationships between words better. BERT can capture the bidirectional context and learn the interaction between words and phrases. 2) solving ambiguity: There can be some ambiguous expressions in the context. And pretraining enables the model to cultivate a nuanced comprehension of language, and makes it adapt to navigate scenarios where identical words may convey distinct emotions contingent upon the surrounding context. 3) enriched feature representations: The layers of the pre-trained models can capture hierarchical features of the datasets, which is valuable for emotion prediction since emotions are often expressed through a combination of linguistic patterns and semantics. 4) transfer learning: Pretraining allows a model pretrained on a substantial corpus to undergo fine-tuned for a specific emotion prediction task using a smaller dataset, which is quite important when labeled emotion datasets are scarce. Leveraging the knowledge acquired during the pretraining stage, the model can enhance its performance on the targeted task by adapting its learned representations to the nuances and intricacies of emotion-related language found in the specific dataset used for fine-tuning (6).

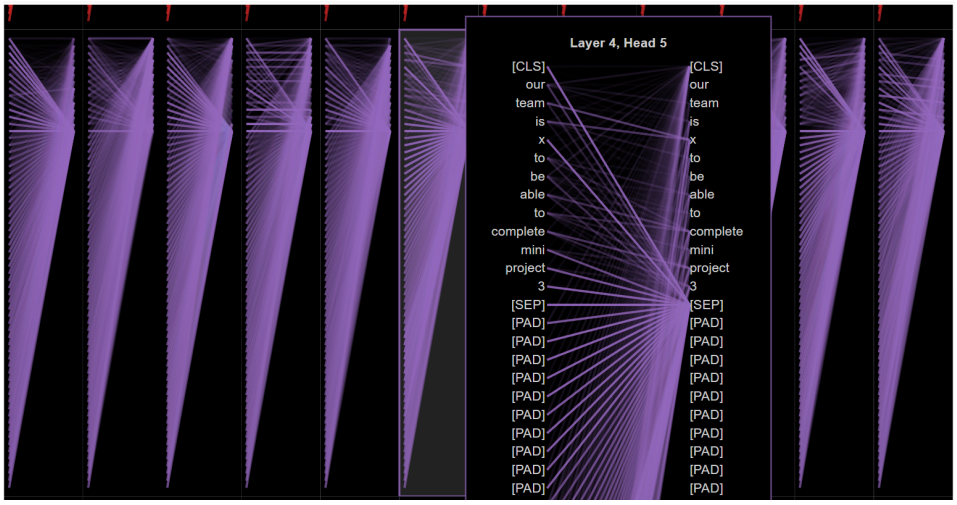

Figure 4: Attention matrix of incorrectly predicted documents.

3.6 Reflections, discussions, and conclusions

In this project, we implement the Naive Bayes model and a BERT-based model on an emotion dataset. From the results, we observed that the accuracy of the BERT-based model outperforms the Naive Bayes model. Pretraining benefits the process in several ways like contextual understanding and handling ambiguity.

However, the computational complexity and cost of the BERT-based model are also higher than the Naive Bayes model, which adds to the cost of experiments. In real applications, we should choose a more suitable one based on specific tasks and features of datasets.

4 Future investigation

There are potential areas for more exploration in this project. Firstly, we could carry out experiments about the performance of different fine-tuned methods. Secondly, when implemented on different datasets, the feasibility of these two methods can switch. Thirdly, we can further explore the performance and traditional machine learning methods and deep learning methods.

5 Statement of Contributions

This project is completed as a team of 3, and the work of this project is rather evenly distributed among the group members. Tian Bai established the model's general structure and framework, while Yijie Zhang ran the experiments and made further adjustments and modifications to the model. Yiping Liu was responsible for the report writing part. Three team members reviewed and finalized the report together.

References

[1] Aggarwal, Charu C., and ChengXiang Zhai. "A survey of text classification algorithms." Mining text data (2012): 163-222.

[2] Pretraining allows the model to develop a nuanced understanding of language, helping it handle situations where the same words might convey different emotions based on context.

[3] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[4] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[5] Hao, Yaru, et al. "Visualizing and understanding the effectiveness of BERT." arXiv preprint arXiv:1908.05620 (2019).

[6] Acheampong, Francisca Adoma, Henry Nunoo-Mensah, and Wenyu Chen. "Transformer models for text-based emotion detection: a review of BERT-based approaches." Artificial Intelligence Review (2021): 1-41.

[7] V. Metsis, I. Androutsopoulos and G. Paliouras (2006). Spam filtering with Naive Bayes – Which Naive Bayes? 3rd Conf. on Email and Anti-Spam (CEAS).

[8] A. McCallum and K. Nigam (1998). A comparison of event models for Naive Bayes text classification. Proc. AAAI/ICML-98 Workshop on Learning for Text Categorization, pp. 41-48.