COMP 551 MINI-PROJECT 3 REPORT

Edmund Wu
261033501

Tianyi Xu
261082272

Santosh Passoubady
261098017

November 17, 2023

ABSTRACT

There exists many machine learning models that are designed to classify data. This mini-project is concerned with one of them, the Naive Bayes classifier. We implement the Naive Bayes classifier on sequential data, the "Emotion" dataset, and then we compare its performance with pretrained BERT-based models. The goal of this mini-project is to assess the effectiveness of deep learning methods, specifically pre-trained models like BERT, in comparison to traditional machine learning approaches like Naive Bayes, for emotion prediction in sequential data.

1 Introduction

One of the many use cases of machine learning models is classification. Given a specific classification problem, we would like a model that ideally always outputs a correct class with high confidence. However, such a model is quite infeasible since our model may be confidently wrong on say, untrained data. The approach of Naive Bayes reduces in a way the confidence by considering a model for each target class. Similar to Bayesian model averaging, Naive Bayes can be seen as keeping as many models as there are of classes. It then combines these models to decide on a final output.

Naive Bayes is advantageous when it is in our interest to keep some form of uncertainty in our model. Not only does it treat the model parameters as random, but it also keeps track of a certain amount of models before mixing them together. Indeed, we are actually mixing models together due to the final Softmax layer, attributing real weights (non-negative and summing to one) to each model.

In this mini-project, we implement a Naive Bayes model that classifies sequential data. We also fine-tune a pre-trained BERT-based model, and then we compare their performances. Specifically, the dataset used is the *emotion* dataset on *huggingface*.

2 System Models

2.1 Naive Bayes Model

At its core, the Naive Bayes model is a probabilistic-driven model implementing the Bayes' Theorem. This model also makes the fundamental assumption of conditional independence among features in our input data. It is also due to this assumption that the Naive Bayes model is able to simplify complex tasks such as emotional detection as it will assume each feature contributes independently to the emotion expressed. The Naive Bayes model works as follows. First, it generates a prior probability for each class. Then, it calculates the likelihood of each piece of data based on said probability. Lastly, it generates a posterior probability distribution that will be used to make future predictions. This approach makes the Naive Bayes model simple but also computationally efficient, making it ideal for tasks such as emotion detection.

Alternative models to the Naive Bayes model are any classification models, including linear or logistic classification, or perhaps even more statistical models such as regression trees. However, as discussed above and in the introduction, Naive Bayes has an advantage even keeping more randomness and uncertainty.

2.2 BERT Model

BERT, also known as Bidirectional Encoder Representation from Transformers, is a family of language models. Unlike the traditional models that process inputs in a linear fashion, BERT models have a bidirectional characteristic in their implementation. This means that BERT considers inputs from both directions (left and right). This factor makes BERT highly effective for complex tasks such as emotion detection. In this assignment, we explored several different BERT models:

- **DistilBERT**
  DistilBERT is, at its core, a streamlined version of the original BERT model and it is used for its faster performance as well as lower resource resource consumption. Compared to BERT, DistilBERT has fewer layers and parameters.
- **BERT Base Uncased Emotion**
  BERT Base Uncased Emotion model is a variant of the original BERT Base model that is specifically fine-tuned for emotion detection tasks. As can be seen from its name, this model does not differentiate the difference between upper and lower cases. This feature proves to be useful in tasks such as emotion detection as text inputs can be extremely inconsistent in capitalization and ignoring such inconsistency could allow the model to learn more information about the word itself rather than capitalization patterns.
- **DistilBERT Base Uncased Emotion**
  DistilBERT Base Uncased is in essence a combination of DistilBERT and BERT Base Uncased Emotion model. It is able to combine the efficiency of DistilBERT alongside the specialized focus of emotion detection on the BERT Base Uncased Emotion model. Also, being an Uncased model, DistilBERT Base Uncased does not separate between an upper and lower case letter. And being a DistilBERT model, DistilBERT Base Uncased inherits a lighter and faster alternative to the full-sized BERT models with only a slight compromise in performance. Lastly, being a BERT emotion model, DistilBERT Base Uncased is specifically fine-tuned to recognize and classify emotions in texts. Now, combining all the qualities above, the DistilBERT Base Uncased Emotion model naturally becomes a particularly suitable choice for emotion-processing tasks.

3 Dataset

The Emotion dataset, provided by DAIR.AI, comprises English Twitter messages, each labeled with six basic emotions: anger, fear, joy, love, sadness, and surprise. The dataset is made of 16,000 training, 2,000 validation, and 2,000 test samples. This dataset is used for multi-class emotion classification in text, aiming to predict the emotional context of tweets.

The following bullet points describe how we analyzed our data set:

- **Data Preparation**
  Loaded the dataset, split it into training and test sets, then transformed the text into a matrix of token counts using `CountVectorizer` and encoded the labels with `LabelEncoder`.
- **Data Exploration**
  Conducted an analysis of word frequencies, label distribution, and document lengths in the dataset to gain insights into common words, the balance of emotion categories, and the verbosity of the texts.

4 Results

The set of parameters that were used consistently for the corresponding models across the experiment are defined in Appendix A.

4.1 Experiment 1: Comparison of Model Accuracy

Table 1: Accuracy on Training and Test Set of Various Models

Set	Multinomial Naive Bayes	`bert-base-uncased-emotion`	`distilbert-base-uncased-emotion`
Training	0.8761	0.9838125	0.9881875
Test	0.7655	0.9265	0.927

The comparison reveals that both BERT and DistilBERT models exhibit high accuracy on the Emotion classification task, with DistilBERT slightly outperforming BERT on both the training and test sets. Despite expectations for the full BERT model to lead due to its larger and more complex architecture, DistilBERT matches its performance, suggesting that the streamlined version of BERT successfully retains the core capabilities while possibly benefiting from greater generalizability.

In contrast, the Multinomial Naive Bayes model, which relies on word frequency, exhibits a notable decrease in accuracy from training to test data, suggesting potential overfitting or insufficient generalization to new data. This shortfall, in contrast to the deep learning models, is likely because its simpler statistical method is less capable of grasping the complexities of natural language.

4.2 Experiment 2: Attention Matrix

We visualize the attention matrix of the output of a DistilBERT model. We have used the library *BertViz*. The following shows the attention matrix of the heads for three correct and three incorrect text inputs, as well as one correct and one incorrect input for the whole model.

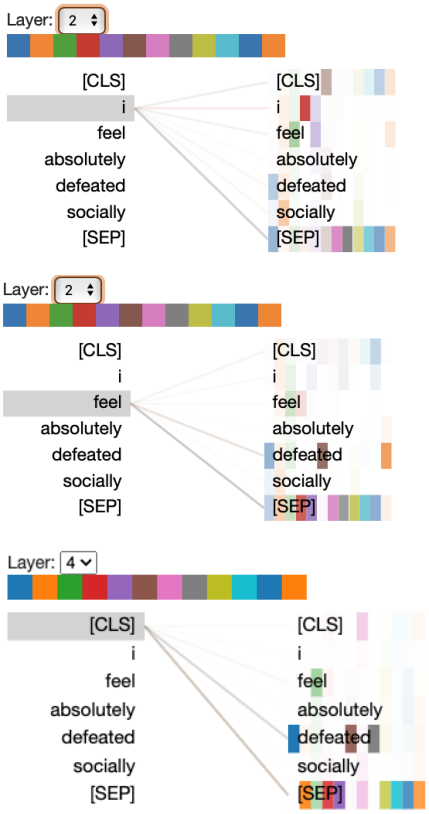

Figure 1: Head View - Correct Input

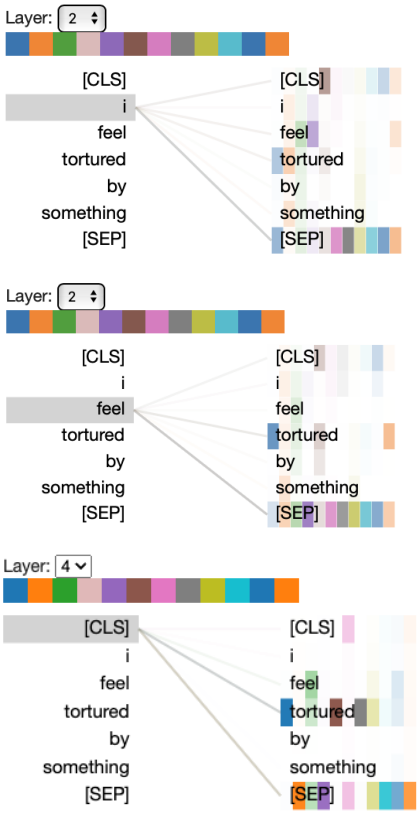

Figure 2: Head View - Incorrect Input

The above figure further justifies how our model is not the best at generalization. In particular, the in third row of attention matrices, we can see that the incorrectly classified input puts little weight on *tortured*, while the correctly classified input puts most weight on *defeated* as it should. Attention weights indicate the influence of a token on the final class. There may be complex interactions, but it seems to make sense that one would expect the words *defeated* and *tortured* to be the main contributing weights that determine the class.

Attention matrices seem to be a useful visual tool to see which target tokens are the ones that the model aims for. However, for long sequences in more complicated models, they may seem to be confusing and perhaps misleading at times. Attention matrices do not seem to be a necessity to understand the dynamics of the model, but rather just a tool that can visualize such dynamics.

4.3 Experiment 3: Fine-tuned Model Comparison

Table 2: Accuracy on Training and Test Set for DistilBERT and Fine-Tuned Variants on our Dataset

Set	`distilbert-base-uncased-emotion`	Fine-Tuning Restricted to Last 2 Layers	Fully Fine-Tuned
Training	0.9881875	0.9909375	0.997
Test	0.927	0.9275	0.932

Our results demonstrate that as we increase the extent of fine-tuning on the DistilBERT model, we observe a corresponding improvement in its ability to classify emotions in sequential data. Starting with a robust baseline accuracy, the model's performance is enhanced marginally when fine-tuning is applied only to the last layers.

However, the most notable improvement is seen when the model undergoes a comprehensive fine-tuning across all layers. This suggests that fine-tuning the model in its entirety allows it to better adapt to the nuances of our specific task, thereby improving its predictive capabilities. These findings support the hypothesis that targeted fine-tuning of pre-trained models on task-specific corpus is a crucial step towards achieving optimal performance.

4.4 Experiment 4: Comparison of Modified Model Structure

Figure 3: Comparison of Training Loss on Our Dataset After 3 Epochs with a Learning Rate of 5e-5: Original vs. Modified Versions of DistilBERT-Base-Uncased-Emotion

Table 3: Accuracy on Test Set of our Dataset for DistilBERT-Base-Uncased-Emotion and its Modified Variants

Model	**Accuracy**
Original Model	0.908
ReLU Activation	0.853
Dropout Rate = 0.2	0.9025
Dropout Rate = 0.3	0.905
Num. Hidden Layers = 8	0.8975
Num. Hidden Layers = 10	0.883
Num. Heads = 8	0.9
Num. Heads = 16	0.8965

- **Original Model**
  This serves as the benchmark with a relatively low training loss that converges just above 0.2 and achieves the highest accuracy on the test set at 0.908. It sets a standard for comparison with other variants.
- **ReLU Activation**
  Introducing ReLU activation results in a higher training loss, indicating a slightly less efficient learning process compared to the original model. Its accuracy on the test set is also notably lower at 0.853, making it the least accurate among the variants.
- **Dropout Rate = 0.2**
  With this modification, the training loss is similar to the original model for the initial steps but finishes slightly higher. The accuracy is marginally lower than the original at 0.9025, suggesting a small impact from the increased dropout rate.
- **Dropout Rate = 0.3**
  This model shows a higher training loss throughout the steps compared to the original model, finishing closer to 0.4. The test set accuracy is close to the original, at 0.905, implying that despite a higher training loss, the test accuracy is not as heavily impacted.
- **Num. Hidden Layers = 8**
  This variant shows a higher training loss across all steps and concludes with a loss near 0.3. Its accuracy on the test set drops to 0.8975, reflecting a decrease in model performance with fewer hidden layers.
- **Num. Hidden Layers = 10**
  Exhibiting a training loss curve with a steeper descent initially but plateauing above the original model, this variant ends with a higher loss. The accuracy on the test set is the lowest at 0.883, suggesting that adding layers did not benefit the model within the tested parameters.
- **Num. Heads = 8**
  The training loss remains consistently above the benchmark model throughout training, indicating less efficiency. However, its accuracy on the test set is quite robust at 0.9, only slightly below the original.
- **Num. Heads = 16**
  This model shows a training loss that is close to the original model's curve, suggesting a good learning process. The test set accuracy is quite high at 0.8965, indicating that the increase in the number of attention heads has a positive impact on the model's ability to generalize.

Overall, while some modified structures show competitive test accuracies, none outperform the original model in both training loss and test set accuracy. This indicates that the original DistilBERT-Base-Uncased-Emotion model structure is quite robust, and modifications such as altering activation functions, dropout rates, or the number of layers and attention heads produce varied effects, with none offering clear superiority over the original configuration in this particular experiment.

We should also note that comparisons between the original DistilBERT model and its variants with altered numbers of hidden layers or attention heads should be approached with caution. These variants, due to their architectural changes, require a longer learning curve to optimize the additional parameters they introduce. Consequently, over a short training duration such as three epochs, these models may not reach the performance level of the original, finely-tuned architecture.

5 Discussion and Conclusion

Through our experiments, it became evident that pretraining on an external corpus, a strategy employed by models like BERT, significantly enhances performance. This enhancement is attributable to the comprehensive learning from a diverse corpus, equipping these models with a profound understanding of language nuances. However, we notice that an important aspect that further boosts their effectiveness is fine-tuning them on data specific to our task. This targeted training allows the models to adapt their broad linguistic knowledge to the particularities and specific requirements of emotion prediction that simpler models may miss.

Comparing deep learning methods like BERT to traditional machine learning approaches such as Naive Bayes reveals a marked superiority of the former in natural language processing tasks. Deep learning models not only achieved higher accuracy but also demonstrated better generalization from training to test data. This contrast highlights the limitations of traditional models in handling the intricacies of language, underlining the need for sophisticated approaches like BERT in tasks requiring a nuanced understanding of text, such as emotion prediction.

Future investigation could focus on improving the explicability and reducing the biases in models such as BERT, as it is essential to understand how these transformer models can inadvertently absorb and manifest societal biases in their results.

6 Statement of Contributions

Each member contributed equally to this project. Each of us completed this lab individually and wrote the report by merging our responses and code.

References

[1] Saravia, E., Liu, H.-C. T., Huang, Y.-H., Wu, J., & Chen, Y.-S. (2018). CARER: Contextualized Affect Representations for Emotion Recognition. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 3687–3697. https://doi.org/10.18653/v1/D18-1404

[2] dair-ai, emotion dataset, https://huggingface.co/datasets/dair-ai/emotion

[3] DistilBERT, https://huggingface.co/docs/transformers/model_doc/distilbert

[4] Prémont-Schwarz, I. (2023). Naive Bayes. COMP 551, McGill University.

[5] Prémont-Schwarz, I. (2023). Neural Networks for Sequences. COMP 551, McGill University.

Appendix A Default Model Parameters

Table 4: Hyperparameters Employed in Fine-Tuning DistilBERT-Base-Uncased-Emotion for Experiment 3

Learning Rate	Batch Size	Epochs
5e-5	16	5

Table 5: Hyperparameters Employed in Training the Variant Model Structure of DistilBERT-Base-Uncased-Emotion for Experiment 4

Learning Rate	Batch Size	Epochs
5e-5	16	3

Table 6: Default hyperparameters of DistilBERT-Base-Uncased-Emotion

Activation	Dropout Rate	Num. Hidden Layers	Num. Heads
GELU	0.1	6	12