Text classification: Emotion Analysis

1 Abstract

The research project explores Emotion dataset-based emotion classification using Naive Bayes [5] and a BERT-based model [3] for understanding emotional content in text [6]. Detecting emotions aids machines in expressing and comprehending emotions within textual data, a vital aspect of emotional intelligence. This study aims to compare traditional Naive Bayes with advanced BERT model, emphasizing the significance of fine-tuning [4] pre-trained BERT models.[4] The findings strongly favor the BERT model due to its ability to extract semantic nuances from the data. This research provides valuable insights into applying machine learning algorithms for emotion detection in text, bridging classical approaches with advanced models for enhanced comprehension of emotional content.

2 Introduction

In this study, we delve into the complexities of emotion analysis using the Emotion dataset curated by dair-ai, employing two distinctive models: Naive Bayes [5] and the state-of-the-art BERT model [3]. Our initial exploration involved applying Gaussian Naive Bayes and Multinomial Naive Bayes, revealing the latter's superior performance due to the dataset's deviation from a Gaussian distribution. Subsequently, we conducted a comparative analysis between Naive Bayes and BERT. Delving deeper, we embarked on the intricate task of fine-tuning the BERT model, initially focusing on fine-tuning the last two layers, and subsequently, fine-tuning all layers. The outcomes unveiled a remarkable 90% accuracy for the finely tuned BERT model, showcasing its exceptional ability to comprehend and classify emotions within the dataset. In contrast, the Naive Bayes model achieved an accuracy rate of 56%, as expected, given Naive Bayes' suitability for smaller datasets owing to its strong independence assumption property. This discerning analysis unequivocally establishes the BERT model as the superior performer for the Emotion dataset in our study. Our comprehensive exploration of implementation, fine-tuning, and evaluation contributes to a nuanced understanding of these models' strengths and capabilities in the realm of emotion detection. Additionally, our analysis includes extensive visualizations, graphs, and tables, providing a thorough examination of machine learning algorithms on the emotion dataset.

3 Datasets

In the exploration of the Emotion dataset, we employed a comprehensive set of visualizations and preprocessing methods. Leveraging the CountVectorizer [1] and N-gram [2] distribution analysis, we specifically focused on unigrams and bigrams to extract meaningful insights from the text data. The visualizations presented in Figure 1 and Figure 2, which include bar plots featuring the top 20 frequently occurring single and two-word phrases, offered detailed insights into the distinct linguistic characteristics associated with various emotions, such as sadness, joy, love, anger, fear, and surprise. Accompanied by word clouds that visually represented the distribution of words as shown in figure 3 and figure 4, our analyses unveiled distinctive language patterns and expressions within the dataset. We conducted exploratory analysis, exemplified by evaluating class distribution shown in Figure 5, to enhance our understanding of the data.

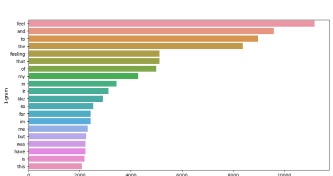


Fig.1 Top 20 1 Gram

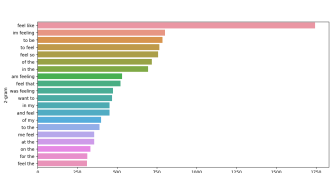


Fig.2 Top 20 2 Gram

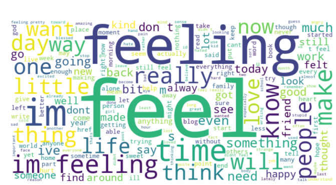


Fig.3 1-Gram Word Cloud

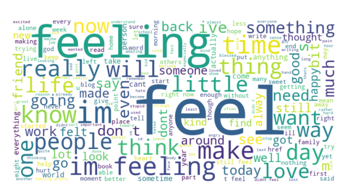


Fig.4 2-Gram Word Cloud

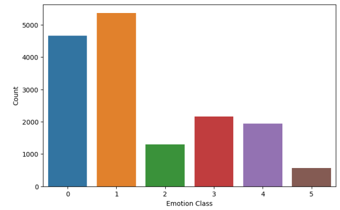
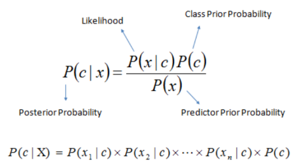


Fig.5 Class Distribution

4 Models

4.1 Naive Bayes

In the Models section of our project, the Naive Bayes model stands out as a probabilistic algorithm based on Bayes' theorem, characterized by its simplicity and efficiency. The model assumes independence among features, allowing it to make predictions swiftly with relatively few parameters.



$$P(c \mid x) = \frac{P(x \mid c) P(c)}{P(x)}$$

$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Fig.6 Naive Bayes Equation

In the context of emotion classification, Naive Bayes excels by leveraging probabilities to assign the most likely emotion label to a given input based on the observed features. Through meticulous training and optimization, we harness the power of Naive Bayes to enhance the accuracy and interpretability of our emotion classification model.

4.2 BERT

In this project we use BERT (Bidirectional Encoder Representations from Transformers) [3] which is one of the important model in Natural Language Processing field. The important aspect of this model is using a bidirectional transformer architecture which utilizes self attention to process the input tokens (text) in a non-linear fashion which helps them to consider long term dependencies of input tokens unlike RNN [7] which fails to do that. We use a pre-trained model which is trained on a large dataset and This pre-training endows BERT with a deep understanding of language intricacies, which is then fine-tuned [4] for specific NLP tasks with additional task-specific layers. This approach allows BERT to achieve remarkable accuracy in tasks such as emotion classification outperforming traditional Naive Bayes .

```
BertForSequenceClassification(
  (bert): BertModel(
    (embeddings): BertEmbeddings(
      (word_embeddings): Embedding(30522, 768, padding_idx=0)
      (position_embeddings): Embedding(512, 768)
      (token_type_embeddings): Embedding(2, 768)
      (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
      (dropout): Dropout(p=0.1, inplace=False)
    )
    (encoder): BertEncoder(
      (layer): ModuleList(
        (0-11): 12 x BertLayer(
          (attention): BertAttention(
            (self): BertSelfAttention(
              (query): Linear(in_features=768, out_features=768, bias=True)
              (key): Linear(in_features=768, out_features=768, bias=True)
              (value): Linear(in_features=768, out_features=768, bias=True)
              (dropout): Dropout(p=0.1, inplace=False)
            )
            (output): BertSelfOutput(
              (dense): Linear(in_features=768, out_features=768, bias=True)
              (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
              (dropout): Dropout(p=0.1, inplace=False)
            )
          )
          (intermediate): BertIntermediate(
            (dense): Linear(in_features=768, out_features=3072, bias=True)
            (intermediate_act_fn): GELUActivation()
          )
          (output): BertOutput(
            (dense): Linear(in_features=3072, out_features=768, bias=True)
            (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
            (dropout): Dropout(p=0.1, inplace=False)
          )
        )
      )
    )
    (pooler): BertPooler(
      (dense): Linear(in_features=768, out_features=768, bias=True)
      (activation): Tanh()
    )
  )
  (dropout): Dropout(p=0.1, inplace=False)
  (classifier): Linear(in_features=768, out_features=6, bias=True)
)
```

Fig.7 BERT Architecture

5 Results

5.1 Naive Bayes

Two Naive Bayes models [5], Gaussian Naive Bayes and Multinomial Naive Bayes, were evaluated on the Emotion dataset. The performance of the models was assessed based on their accuracy.

5.1.1 Gaussian Naive Bayes

The maximum achieved accuracy using the Gaussian Naive Bayes model was 36.8%. Attempts to optimize the model by tuning the Laplace smoothing parameter (alpha) across a range of values (1 to 10) did not yield significant improvements. This aligns with the dataset's inherent distribution, which does not follow a Gaussian distribution, thereby limiting the efficacy of this model for this particular dataset.

5.1.2 Multinomial Naive Bayes

In contrast, the Multinomial Naive Bayes model [5] outperformed the Gaussian variant, achieving an accuracy of 56%. Further fine-tuning [4] of the Laplace smoothing parameter (alpha) within the range of [1, 5, 7, 9] revealed that an alpha value of 1 yielded the highest accuracy of 56%. This demonstrates the suitability of the Multinomial Naive Bayes approach for this dataset. The results are illustrated in Table 1.

alpha	accuracy	Precision	Recall	F1-score
1	0.5665	0.5665	0.5665	0.5665
5	0.5114	0.5114	0.5114	0.5114
7	0.5005	0.5005	0.5005	0.5005
9	0.4905	0.4905	0.4905	0.4905

Table 1: Effect of Alpha

The text data was preprocessed and vectorized using a unigram approach (n_gram range of (1,1)) before being fed into the models for analysis.

5.2 BERT

In this section, we show analysis on 3 different flavors of BERT model [3] which are pre-trained model, fine-tuned on few layers and fine-tuned [4] on all layers and performed analysis on dair-emotion dataset.

5.2.1 Pre-trained BERT model

In this section of our report, we used the BERT model that was already trained on emotion-related data from Twitter. This means the model already knew a lot about language as it's used on Twitter. Using a model that's already trained like this is a good idea because it saves us time and effort, and it usually works well, especially if our data is similar to what the model was trained on. We discovered that this pre-trained model worked well with our dair-emotion dataset, showing its ability to apply what it learned from Twitter to a different set of data. We achieved an accuracy of 92% with this approach.

5.2.2 BERT Finetuned on last few layers

Here, we used BERT model, specifically used fine-tuning [4] its last two layers for the task of emotion classification. We fine-tuned last 2 layers of the BERT using the training dataset which consists of various samples, each tagged with specific emotional labels. The reason we do fine-tuning on last few layers is to make these layers more adaptable to downstream tasks i.e emotion classification while initial layers retain their general understanding of the language. After fine-tuning, we then evaluate the models's performance on test dataset. The finetuned version performed better than the Naive Bayes approach. Fig 8 is the heatmap of attention matrix for correctly predicted at layer number 4, Fig 9 is for layer number 7 and Fig 10 is for layer number 10. Fig 11. shows the heatmap of attention matrix of incorrectly predicted sample at layer number 4 and Fig 12 and Fig 13 at layer number 7 and 10 respectively.

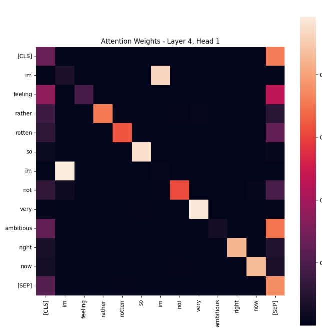


Fig.8 Correctly Predicted Layer 4

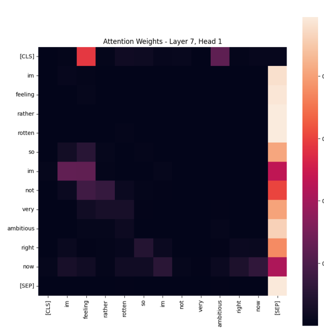


Fig.9 Correctly Predicted Layer 7

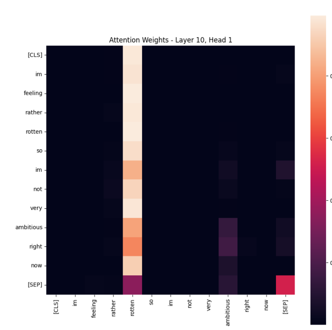


Fig.10 Correctly Predicted Layer 10

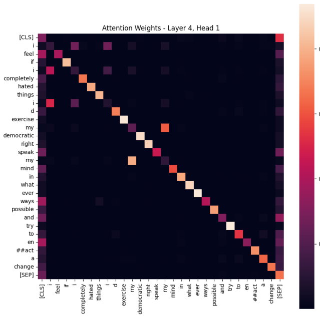


Fig.11 Incorrectly Predicted Layer 4

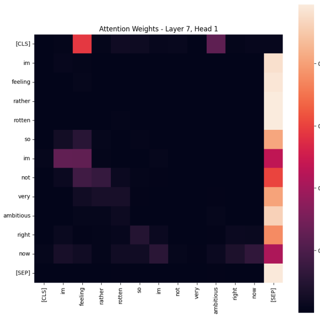


Fig.12 Incorrectly Predicted Layer 7

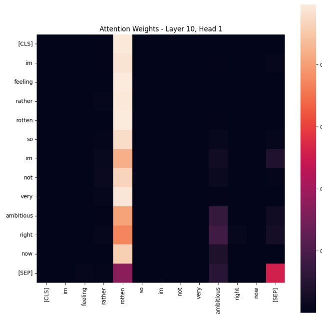


Fig.13 Incorrectly Predicted Layer 10

5.2.3 BERT finetuned on all layers

In this part of our project, we used the BERT model [3] and fine-tuned [4] all its layers for the emotion classification task. We chose to fine-tune all the layers because we wanted to adjust the initial layers of the model as well. This approach meant that not only the top layers, which are usually more specific to the task, but also the base layers of BERT, which are more about general understanding of language, were tailored to our needs. By doing this, we aimed to make the entire model better suited for identifying and classifying emotions in text. This method allows the model to learn and adapt fully to the specifics of our emotion classification task. Fig 14 is the heatmap of attention matrix for correctly predicted at layer number 4, Fig 15 and Fig 16 is for layer number 7 and 10 respectively. Fig 17. shows the heatmap of attention matrix of incorrectly predicted sample at layer number 4 and Fig 18 and Fig 19 at layer number 7 and 10 respectively.

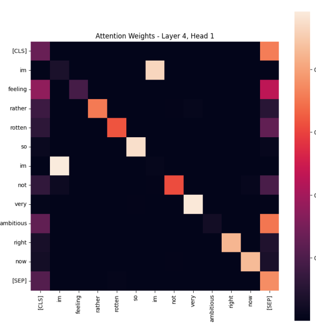


Fig.14 Correctly Predicted Layer 4

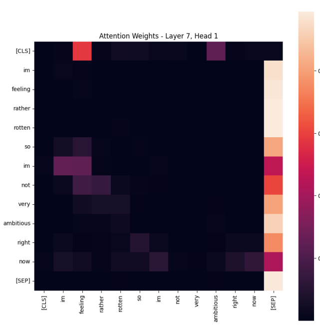


Fig.15 Correctly Predicted Layer 7

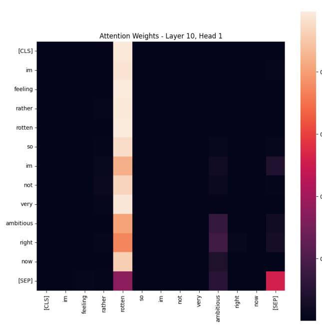


Fig.16 Correctly Predicted Layer 10

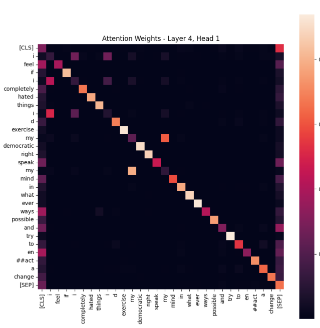


Fig.17 Incorrectly Predicted Layer 4

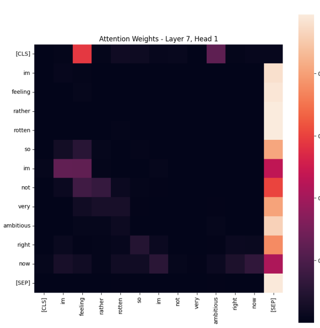


Fig.18 Incorrectly Predicted Layer 7

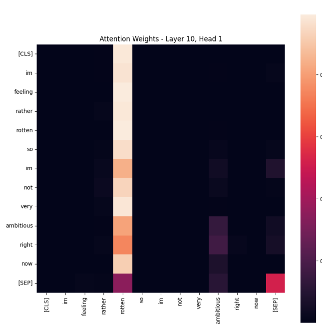


Fig.19 Incorrectly Predicted Layer 10

5.3 Is pretraining on an external corpus (like BERT does) good for the Emotion prediction task?

Pretraining on an external corpus, such as what BERT [3] is beneficial for emotion prediction task. The first advantage of pretraining is that it allows the model to develop a comprehensive understanding of language before it is fine-tuned [4] for a specific task like emotion prediction. The core knowledge includes understanding of syntax, semantics, and the context in which words are used, which is crucial for accurate prediction. For predicting emotions, it's really important to understand

Parameter	Naive Bayes	BERT (FT All Layers)	BERT (FT Few Layers)	BERT (Original Model)
Accuracy	0.5665	0.929	0.9255	0.9265
F1 Score	0.5665	0.929	0.9255	0.9265
Recall	0.5665	0.929	0.9255	9265
Precision	0.5665	0.931	0.9257	0.9267

Table 2: Benchmark of Results

the small details and special ways people use language. A model like BERT, which has been trained on a huge variety of texts from the internet, learns about many different ways people show emotions in their words. This training helps the model to notice and understand even the little signs of emotions in what people write, making it better at figuring out the emotions, even when they are not clearly stated.

5.4 Conclusion about performance between Deep Learning and Traditional Machine Learning

The comparison between deep learning methods like BERT [3] and traditional machine learning approaches such as Naive Bayes shows a clear performance difference in favor of deep learning for tasks like emotion classification. Traditional methods, while effective for simpler tasks, often fall short in capturing the complexities of human language, especially when it comes to understanding context and nuanced emotional expressions. Deep learning models, on the other hand, especially those utilizing transformer architectures, are more adept at handling these complexities. Their ability to process and learn from vast amounts of data allows them to develop a more nuanced understanding of language. Additionally, features like attention mechanisms in transformers enable these models to focus on relevant parts of the text, further enhancing their accuracy in tasks like emotion prediction.

6 Discussion and Conclusion

Our exploration of the Emotion dataset highlighted the critical role of model selection and adaptation in achieving superior performance in text classification tasks. The initial comparison between Traditional Naive Bayes (NB) models [5], specifically the Gaussian and Multinomial variants, revealed the limitations of Naive Bayes in handling datasets that deviate from Gaussian distributions. While the Multinomial Naive Bayes model showed better adaptability to the dataset, it fell short compared to the state-of-the-art BERT model [3]. BERT, known for its prowess in capturing contextual nuances within textual data, emerged as a formidable contender, significantly outperforming Naive Bayes models. This outcome was anticipated given Naive Bayes' reliance on strong independence assumptions, making it better suited for smaller datasets.The exploration extended to leveraging a pre-trained BERT model, demonstrating initial promising results due to its pre-existing understanding of language context. Subsequent fine-tuning of the last two layers enhanced the model's adaptability to the specific classification task, surpassing the Naive Bayes model's performance.However, the pinnacle of our study was the fine-tuning [4] of all layers of the pre-trained BERT model, resulting in an exceptional accuracy of 93%. This comprehensive adaptation process underscored BERT's capability to grasp and leverage complex relationships within the Emotion dataset.

In conclusion, our research advocates for the adoption of advanced models like BERT, especially when dealing with intricate datasets like Emotion that demand a deep understanding of contextual dependencies for accurate classification. The limitations of Naive Bayes models, particularly in scenarios with non-Gaussian distributions, highlight the necessity of employing models that can capture nuanced relationships within textual data. The remarkable performance of the fully adapted BERT model signifies its potential as a robust solution for text classification tasks, setting a high benchmark for accuracy and adaptability in similar endeavors.

7 Statement of Contribution

The research paper represents the combined efforts of Shwetali Shimangaud, implementing the Naive Bayes model; Dheeraj Vattikonda, developing and experimenting with BERT; and Chaitanya Tekane, conducting exploratory data analysis and creating informative visuals. Together, their contributions have led to a thorough analysis showcasing the strengths and limitations of text classification models on the Emotion dataset.

References

[1] Tamal Ahmed, Shawly Folia Mukta, Tamim Al Mahmud, Sakib Al Hasan, and Md Gulzar Hussain. Bangla text emotion classification using lr, mnb and mlp with tf-idf countvectorizer. In *2022 26th International Computer Science and Engineering Conference (ICSEC)*, pages 275–280, 2022.

[2] Peter F Brown, Vincent J Della Pietra, Peter V Desouza, Jennifer C Lai, and Robert L Mercer. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–480, 1992.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

[5] Irina Rish. An empirical study of the naïve bayes classifier. *IJCAI 2001 Work Empir Methods Artif Intell*, 3, 01 2001.

[6] Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[7] Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*, 2017.