Yelp review dataset sentiment analysis

*

Shaunak Hemant Joshi
Department of Computer Science & Engineering
Texas A&M University
shaunakjoshi@tamu.edu

***Abstract*—In this report, we leverage transformer architectures, specifically the pre-trained BERT model, to conduct sentiment analysis on a subset of the extensive Yelp review dataset, which comprises approximately 174,000 reviews with associated star ratings. The primary objective is to implement a robust Transformer model for sentiment analysis, utilizing the textual content of reviews and their corresponding star ratings. Our approach involves fine-tuning the pre-trained BERT model on our dataset to enhance its performance in capturing sentiment-related features. This report contributes to the ongoing exploration of transformer models in natural language processing and sentiment analysis, offering insights into their effectiveness when applied to specific domains such as online reviews.**

***Index Terms*—transformers, BERT, sentiment analysis,**

I. Introduction

User feedback forms a crucial element on various online platforms such as Amazon, Walmart, Yelp, Google, and Facebook. These platforms provide users with the opportunity to share their opinions through written reviews accompanied by star ratings. Reviews typically consist of textual entries where users express their sentiments and experiences regarding a product or service.

The star ratings, ranging from 0 to 5, serve as a quantifiable measure of user satisfaction, with 0 indicating a negative experience and 5 denoting an exceptional one. This feedback mechanism is invaluable for prospective users seeking insights into the quality of products or services. Additionally, businesses can leverage this feedback to enhance their offerings, addressing specific areas of improvement based on user opinions. The interplay between user reviews and star ratings fosters a dynamic ecosystem that benefits both consumers and businesses alike.

Moreover, consumer feedback affects more than just individual purchases; it also shapes broader market trends and a company's reputation. In addition to attesting to the excellence of a good or service, a favorable review helps establish the brand's credibility and trustworthiness. On the other hand, unfavorable reviews might force companies to fix particular issues, proving their dedication to client pleasure. Reviews and star ratings, which represent the aggregate opinion of users, create a vibrant and open online community by giving customers useful information and motivating companies to keep improving their products.

When making purchasing decisions, users often prioritize star ratings over the detailed text reviews. Whether selecting a new restaurant or exploring a new product, users commonly sort options based on star ratings in descending order and opt for those with higher ratings. Despite this common practice, relying solely on star ratings may not accurately capture the nuanced sentiments expressed in the accompanying text reviews. The potential disparity in star ratings assigned by different users to the same textual content highlights the subjective nature of this metric.

To address this inherent bias, a sentiment categorization approach is proposed: star ratings greater than or equal to 4 are deemed positive, those less than or equal to 2 are considered negative, and a rating of 3 is labeled as neutral. This categorization into "Negative," "Positive," and "Neutral" sentiments allows for a more nuanced analysis of user opinions, aligning with the goal of predicting sentiment based on the review text. This approach aims to provide a more comprehensive understanding of user sentiments beyond the limitations of numerical star ratings.

In light of the limitations of relying solely on star ratings, the process of sentiment categorization offers a more refined understanding of user opinions. By categorizing star ratings into three sentiments - "Negative," "Positive," and "Neutral" - users and businesses can glean deeper insights into the collective sentiment expressed in reviews. This approach not only mitigates the subjectivity inherent in numerical ratings but also facilitates a more nuanced analysis of the sentiments conveyed in the accompanying text. Users benefit from a more comprehensive evaluation of products or services, enabling them to make informed decisions beyond the constraints of a numerical scale. For businesses, this approach provides valuable feedback for strategic improvements and a more accurate representation of customer satisfaction, ultimately contributing to enhanced user experiences and product/service enhancements.

This report delves into the endeavor of predicting sentiment in review texts utilizing the Yelp Review Dataset. Leveraging

both the textual content and star ratings from the Yelp dataset, we embark on a multi-class classification task, categorizing sentiments into three classes: "Positive," "Negative," and "Neutral." This classification paradigm enables a more nuanced understanding of user sentiments compared to a traditional star rating approach. The methodology involves employing the BERT pretrained transformer and fine-tuning it to align with the nuances of the Yelp Review dataset.

In Section 2, we delve into the current state of the art in sentiment analysis, examining key methodologies and advancements in the field. Section 3 offers a detailed exploration of transformers, BERT model, elucidating their architecture and functionality. Following this, Section 4 outlines the pre-processing steps undertaken to prepare the Yelp Review dataset for the sentiment classification task. The subsequent sections meticulously detail the implementation of transformers (Section 5) and present the obtained results, accompanied by insightful analysis (Section 6). The report concludes with summarizing remarks and outlines directions for potential future work in Section 7.

II. Related Work

This section includes the previous work, recent advancements, performance metrics used for evaluations and the challenges faced while coming up with these solutions.

A. Previous Work

In [8], Siqi Liu, conducts an ablation study on text preprocessing techniques and assesses the performance of various machine learning and deep learning models in predicting user sentiments. For machine learning models, incorporating binary bag-of-word representation, bi-grams, minimum frequency constraints, and text normalization positively impacts performance. In the case of deep learning models, leveraging pre-trained word embeddings and capping maximum length proves beneficial.

Notably, simpler models like Logistic Regression and SVM demonstrate superior effectiveness in predicting sentiments, considering both performance and training time. Simpler models also offer enhanced interpretability, while more complex models necessitate additional techniques, such as LIME analysis, for understanding their inner workings.Despite the observed advantages of simpler models, deep learning models, with their computational demands, may benefit from improved hardware setups to relax constraints on maximum length and potentially utilize larger datasets.

In the initial stages of sentiment analysis applied to Yelp reviews, conventional machine learning methods took precedence. In a study by S & Ramathmika [1], a supervised machine learning approach was employed to categorize Yelp reviews into graded classifications of positive and negative sentiments. The classification process involved assigning probabilities of goodness or badness based on the occurrence of each adjective within the review text.

Various algorithms, including Logistic Regression, Naive Bayes, Multinomial Naive Bayes, Bernoulli Naive Bayes, and Linear Support Vector Clustering from the Natural Language Toolkit (NLTK) library in Python, were explored in this study. While Naive Bayes exhibited the highest accuracy at 79.12, it faced challenges in accurately assessing long and descriptive sentences. The study also highlighted the limitations of these models in handling nuanced aspects such as detecting sarcasm.

Despite the achievements of early machine learning approaches, the study underscored the necessity for more sophisticated techniques to address the intricacies of sentiment analysis. The limitations in accurately capturing sentiment from longer sentences and discerning subtleties like sarcasm indicated the need for a more nuanced approach. As a result, the study concluded by suggesting a shift in focus towards analyzing reviews based on different features rather than relying solely on the overall sentiment derived from positive and negative adjectives within the reviews. This recommendation laid the groundwork for exploring more advanced methods, including the utilization of deep learning techniques and transformer models, which have since become prevalent in sentiment analysis tasks.

[10] introduces the transformer model, a revolutionary architecture designed for sequence-to-sequence tasks. In contrast to conventional models reliant on recurrent or convolutional layers, the transformer exclusively leverages self-attention mechanisms. The authors posit that this unique attention mechanism enhances the model's ability to capture long-range dependencies more efficiently and allows for parallelization. The pivotal methods introduced in the paper encompass the self-attention mechanism and the multi-head attention mechanism. These mechanisms empower the model to attend to distinct positions using diverse learned linear projections.

Additionally, the incorporation of positional encoding provides crucial positional information about tokens within the input sequence. The transformer architecture, introduced in this paper, has since become a cornerstone in various natural language processing tasks. It has laid the groundwork for subsequent models such as BERT, GPT, and others, demonstrating its enduring impact on the field.

B. Recent Work

Recent advancements in sentiment analysis have embraced the utilization of contextual embeddings and pre-trained language models, marking a notable shift in approach. Notably, fine-tuning models like BERT or GPT on Yelp Reviews has demonstrated superior performance in sentiment analysis. A study conducted by Santiago González-Carvajal

and Eduardo C. Garrido-Merchán [2] stands as a testament to the efficacy of BERT-based models compared to traditional NLP approaches. In their research, González-Carvajal and Garrido-Merchán compared a BERT model against a traditional TF-IDF vocabulary fed to machine learning algorithms, employing accuracy as the benchmark for model performance. Across four distinct experiments, the BERT-based model consistently outperformed traditional NLP models in terms of accuracy. Intriguingly, the study also highlighted that implementing a BERT-based model proved to be less intricate than traditional NLP models.

In [11], we conducted an extensive evaluation of various models for tweet emotion classification using three datasets. Our proposed approach involved hybridizing pre-trained BERT-based classifier models (RoBERTa and DistilBERT) with Bidirectional Gated Recurrent Units (BiGRU) and Bidirectional Long Short-Term Memory (BiLSTM) networks. The preprocessing steps included removing specific elements such as names, trailing whitespace, hashtags, and numbers, followed by tokenization to generate input ids and attention masks for each text line. RoBERTa models without emojis exhibited strong performance when combined with our proposed method.

In summary, the GLG hybridization method proved effective for DistilBERT across all datasets without emojis and for both large and medium datasets with emojis. For RoBERTa, models featuring BiGRU layers outperformed others, particularly for large and small datasets. The incorporation of BiGRU layers with DistilBERT and RoBERTa consistently enhanced accuracy.

In the research conducted by Himanshu Batra, Narinder Singh Punn, Sanjay Kumar Sonbhadra, and Sonali Agarwal [3], three distinct methodologies utilizing the BERT model were applied to scrutinize sentences from GitHub comments, Jira comments, and Stack Overflow posts. The evaluation metric employed for all approaches was the F1 score. In the initial approach, the researchers fine-tuned a BERT-based pretrained model. Subsequently, they adopted an ensemble approach based on the BERT model for the second methodology. Finally, the third approach involved utilizing a compressed BERT model, specifically Distil BERT. The findings of the study underscored the notable advancements achieved by the BERT-based ensemble approach and the compressed BERT model in comparison to other existing tools.

III. Implementation

A. Data Preprocessing

The preprocessing pipeline for the training dataset precedes the application of the BERT model, optimizing it for sentiment analysis. Initial steps involve the removal of punctuation and stop words from the review text. The rationale behind eliminating punctuation lies in its semantic neutrality; punctuation marks don't inherently convey sentiments, and their removal streamlines focus on sentiment-bearing words. Concurrently, the exclusion of stop words, frequently occurring yet semantically less significant words, diminishes noise and enhances the salience of meaningful words. This not only sharpens the sentiment analysis but also mitigates computational overhead, given that stop words contribute minimally to sentiment determination.

Furthermore, all review text undergoes a lowercase conversion, fostering data standardization and normalization. This ensures uniform treatment of words, irrespective of their original casing, facilitating consistent sentiment identification. Standardization proves instrumental in aligning variations like "Great" and "great," treating them as identical entities. The resultant uniformity in sentiment assessment minimizes discrepancies originating from case variations. Additionally, lowercase standardization promotes compatibility across diverse text processing tools and algorithms, streamlining the sentiment analysis workflow.

Addressing the inherent limitation of star ratings as nuanced indicators of sentiment, a categorical transformation is executed. Star ratings are stratified into three distinct categories: "Positive," "Negative," and "Neutral." Ratings surpassing 3 fall into the positive category, those below 3 are assigned to the negative category, and ratings equating to 3 are categorized as neutral. This categorical redefinition aims to foster a more nuanced and accurate sentiment prediction based on the textual content, overcoming the subjectivity inherent in star ratings.

B. Input Data Preparation and Data Exploration

Post data preprocessing, a comprehensive exploration of the dataset's sentiment distribution is conducted. A bar plot, depicted in "Fig. 1" provides a visual representation of sentiment prevalence within the training dataset. Evidently, positive sentiments dominate the dataset, constituting approximately 120,000 text reviews. In contrast, the count for negative sentiment reviews totals around 40,000, while neutral sentiment reviews amount to approximately 20,000.

This stark class imbalance is apparent, with positive sentiments substantially outweighing the combined count of negative and neutral sentiments. The observed class imbalance raises concerns about potential bias in the model towards the majority class, i.e., positive sentiments. The model's tendency to minimize errors might result in less effective learning of nuances and patterns within the minority classes, namely negative and neutral sentiments. Such imbalance poses a risk of poorer generalization and decreased performance on these minority classes when the model encounters new or unseen data.

Before, we move to our model, let's look in detail at the transformers and the pre-trained Bert model that we are using here.

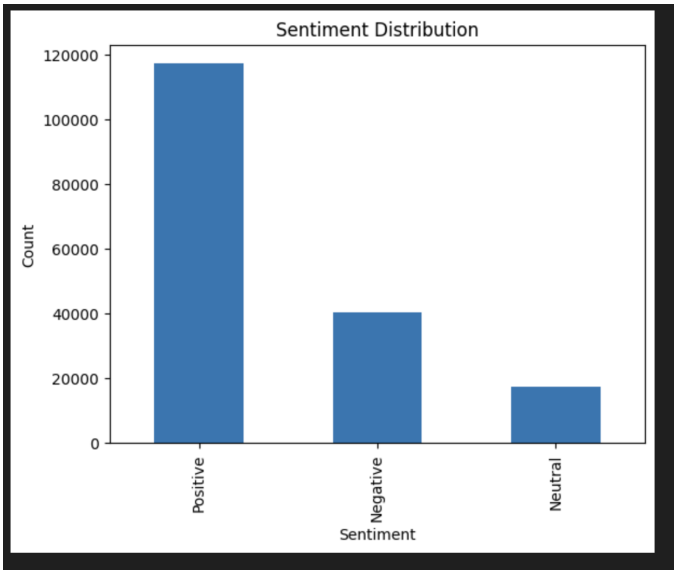

Fig. 1. Class Distribution

C. Transformer Model

A Transformer model is a type of neural network architecture introduced by Vaswani et al. in [10]. Transformer's architecture is based on the self-attention mechanism, which enables it to parallelizably capture the links between words in a sequence. The self-attention mechanism enables the model to effectively capture long-range dependencies by weighing the significance of various words in a sequence with respect to a specific word. It can be used in Natural Language Processing (NLP), Speech Recognition, Image Processing, Time Series Analysis. Next, we move on to a specific transformer model that is well suited to our task here.

D. Tokenization using a pre-trained Bert model

BERT, which stands for Bidirectional Encoder Representations from Transformers, is a transformer-based neural network architecture introduced by Devlin et al. in 2018. Unlike traditional language models that read text sequentially, BERT reads the entire input sentence bidirectionally. This bidirectional context allows BERT to capture the intricate dependencies between words and understand the context in which each word appears.

The pre-training of BERT involves unsupervised learning on massive amounts of textual data, which enables the model to learn rich, context-aware representations of words. Fine-tuning BERT on specific sentiment analysis tasks further refines its ability to understand and classify sentiments in a more targeted manner.

The BERT-base model comprises an encoder consisting of 12 Transformer blocks, each equipped with 12 self-attention heads and a hidden size of 768. When processing input, BERT accommodates a sequence of up to 512 tokens and generates a representation for the entire sequence. This sequence typically consists of one or two segments, with the initial token always designated as [CLS], encompassing a specialized classification embedding. Another distinct token, [SEP], is employed to demarcate segments within the sequence.

Here, we are using a pretrained Bert model, specifically the "bert-base-uncased" model to fine tune our predictions. The "uncased" in its name indicates that the model is trained on lowercase text and is case-insensitive. It is a base version, meaning it has a moderate size with 12 layers, 110 million parameters, and a hidden size of 768. This model has achieved state-of-the-art results in various natural language processing tasks, including text classification, sentiment analysis, and question-answering, among others. BERT models have a maximum input sequence length, typically 512 tokens. If the input sequence is longer, it is truncated, and if it is shorter, padding tokens are added. Padding tokens do not contribute to the meaning but help maintain a consistent input size.

The BERT tokenizer is used to convert text into tokenized sequences, and the resulting token lengths are visualized to gain insights into the dataset's characteristics. In summary, tokenization in BERT involves breaking down text into subword units, adding special tokens, handling padding and truncation, and creating token IDs and attention masks to prepare the input for the model. This process ensures that the input text is properly formatted for BERT's bidirectional context understanding.

E. Our Model

Here, we are making use of three models which we are building it on the top of the fine-tuned pre-trained BERT model "bert-base-uncased.

In the First model, incorporated into the pretrained BERT model are dropout layers aimed at regularization, linear layers for dimensionality reduction, ReLU activation functions to introduce non-linearity, and a concluding linear layer tailored for sentiment classification. The optimization process employs Stochastic Gradient Descent (SGD) with a learning rate set at 0.005. The model undergoes training for a span of 5 epochs. Specifically, dropout rates of 0.4 and 0.3 are implemented within the network. To streamline the architecture, an additional layer is introduced in the fully connected network, strategically reducing the layer count from 128 to 64 in the final stage and then reducing it from 64 to 3 classes. This augmentation enhances the model's capacity for

sentiment analysis and improves its generalization capabilities.

Similarly, in the second model, we have changed the dropout rates to 0.4 and 0.4. Here, we have removed the final layer that reduces the number of layers from 128 to 64. This is done so that the model complexity is reduced and the model does not overfit, so that it can generalize well on unseen data.

In the third model, we have reduced the learning rate to 0.001 A small learning rate allows for more precise adjustments to the model parameters. Small learning rates help prevent overshooting the minimum of the loss function. If the learning rate is too large, the optimization algorithm might oscillate around the minimum or even fail to converge.

The training loop for every model iterates over the training dataset, performs forward and backward passes through the model, computes the loss, and updates the model parameters. Gradient clipping is applied to prevent exploding gradients. The model's performance is evaluated on a validation dataset after each epoch. The evaluation loop calculates accuracy and average loss. Training and validation accuracies and losses are recorded after each epoch for later analysis. The model with the highest validation accuracy is saved, and its checkpoint includes the model state, epoch, and accuracy. The best model's index and accuracy are tracked for reference.

IV. Results and Analysis

As stated above, the third model is the one with the highest validation accuracy and incidentally it also has the lowest validation loss. This might be due to the lower learning rate that we have used here as compared to the second model. A smaller learning rate allows the model to reach the minimum slowly (converge slowly) and then prevents generalization very easily.

After training all the three models, we get the training, validation accuracy and the training, validation loss for every model. We are storing the best accuracy and the best model so that we can load this model later to evaluate on the test dataset. Here, we see that our third model performs the best and it has the largest accuracy on the validation dataset. We, then plot the training loss vs validation loss in "Fig. 3" and the training accuracy vs training accuracy "Fig. 2" for this model.

We have further made prediction on some sample inputs using our model here to further strengthen the confidence in the model.

Post this, we load the best saved model and then perform predictions on the test dataset. On evaluation, we can see that the accuracy on the test dataset comes out to be 0.8732. We can also use other performance metrics to evaluate our models.

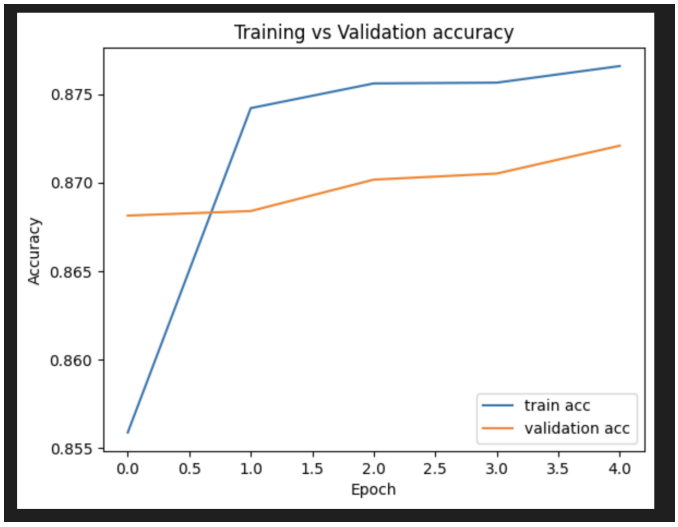

Fig. 2. Training vs Validation Accuracy

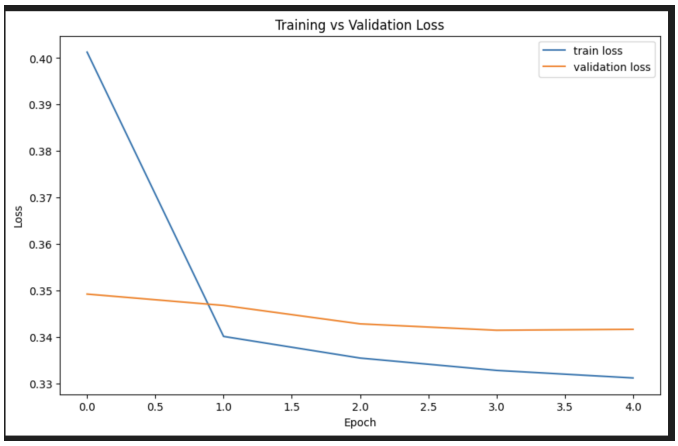

Fig. 3. Training vs Validation Loss

Common sentiment analysis metrics, such as accuracy, precision, recall, and F1-score, gauge the performance of models. Accuracy quantifies correctly predicted sentiments over the total sentiments, a fundamental metric susceptible to imbalanced datasets. Precision gauges the accuracy of positive predictions, emphasizing precision. Recall assesses the model's ability to capture actual positive sentiments. F1-score, the harmonic mean of precision and recall, offers a balanced evaluation.

A diminished precision value for the neutral class signals a high incidence of false positive predictions, reflecting an inadequacy in accurately identifying neutral sentiments. Likewise, a reduced recall value for this class indicates the model overlooks a substantial number of genuine neutral sentiments, underscoring its inadequacy in capturing true neutral sentiments comprehensively. A diminished F1-score, as the harmonic mean of precision and recall, indicates an overall lack of equilibrium between precision and recall,

```
[87]  from sklearn.metrics import classification_report,confusion_matrix
      print(classification_report(y_test, y_pred, target_names=class_levels))

...                 precision    recall  f1-score   support

       Negative       0.84      0.88      0.86      3145
        Neutral       0.47      0.40      0.43      1416
       Positive       0.93      0.94      0.94      9419

       accuracy                           0.87     13980
      macro avg       0.75      0.74      0.74     13980
   weighted avg       0.87      0.87      0.87     13980
```

Fig. 4. Performance Metrics

pointing to suboptimal performance in the accurate prediction of neutral sentiments. On the other hand, the metrics for positive and negative class are very high in value and indicate that our model is good at predicting positive and negative sentiments.

This low performance metrics for the neutral class might be due to the low number of samples belonging to the neutral class in the training dataset as compared to the positive and the negative samples. Due to this reason, the model might not be able to generalize clearly on the test dataset for neutral sentiment as compared to the other two fields in the class.

This can also be seen clearly, if we plot the confusion matrix for this dataset. This matrix provides a comprehensive overview of how well a model performs across different classes, aiding in the evaluation of its accuracy and error patterns. Additionally, the confusion matrix facilitates the identification of specific types of errors, such as instances where the model frequently misclassifies certain classes, enabling targeted improvements in model training and fine-tuning

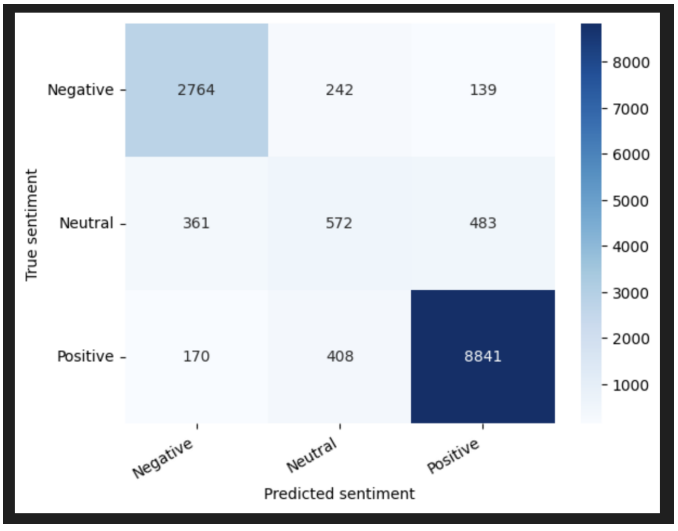

Fig. 5. Performance Metrics

From the confusion matrix, we can see that the number of samples misclassified for positive and negative classes is very low and further strengthens the earlier metrics that the samples in the neutral class are getting misclassified a lot as compared to the other two fields present in the sentiment class. Maybe, we can increase the precision and f-1 score of the misclassified samples by performing data sampling and preprocessing so that the neutral class can get classified in a better way using our model.

V. Conclusion and Future Work

While all three models exhibit robust performance in effectively classifying positive and negative sentiments, they encounter difficulties in accurately discerning neutral sentiment reviews. Despite achieving an impressive overall accuracy of approximately 0.8732, challenges persist in achieving precision within the neutral sentiment category.

Notably, the last two models outperform the initial one in terms of predictive accuracy, which could be attributed to the adoption of a lower learning rate and a reduction in the model's complexity. These adjustments contribute to the development of a model that demonstrates improved generalization capabilities when confronted with previously unseen data, distinguishing it from the earlier two models. This underscores the importance of careful hyperparameter tuning and model architecture adjustments in achieving enhanced performance on sentiment analysis tasks.

The existing imbalance in the dataset, particularly the scarcity of neutral sentiment reviews during training, presents a hurdle for precise classification. In future investigations, potential solutions such as class weighting, data augmentation, and resampling techniques could be explored to address this challenge. Additionally, experimenting with different configurations of transformer architectures, including variations in hidden dimension, number of attention layers and adjustments to dropout, learning rates holds promise for improving the models' performance, especially in the nuanced task of neutral sentiment classification. By implementing these strategies, the goal is to enhance the models' capability to accurately classify sentiments across all categories, taking into account the intricacies introduced by the imbalanced dataset.

References

[1] H. S. and R. Ramathmika, "Sentiment Analysis of Yelp Reviews by Machine Learning,"2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 2019, pp. 700-704, doi: 10.1109/ICCS45141.2019.9065812.

[2] Santiago González-Carvajal and Eduardo C. Garrido-Merchán, "Comparing BERT against traditional machine learning text classification", doi: http://dx.doi.org/10.47852/bonviewJCCE3202838.

[3] Himanshu Batra, Narinder Singh Punn, Sanjay Kumar Sonbhadra, Sonali Agarwal, "BERT-Based Sentiment Analysis: A Software Engineering Perspective doi: https://doi.org/10.48550/arXiv.2106.02581.

[4] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[5] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[6] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[7] How to Fine-Tune BERT for Text Classification? Chi Sun, Xipeng Qiu, Yige Xu, Xuanjing Huang.

[8] Sentiment Analysis of Yelp Reviews: A Comparison of Techniques and Models, Siqi Liu.

[9] https://github.com/skshashankkumar41/Sentiment-Analysis-Using-Transformers-PyTorch

[10] Attention is All you Need Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin (2017)

[11] Talaat, A.S. Sentiment analysis classification system using hybrid BERT models. J Big Data 10, 110 (2023). https://doi.org/10.1186/s40537-023-00781-w