Sentiment Analysis using a Fine-Tuned BERT Model

Arnav Jain

December 12, 2023

Abstract

Sentiment analysis has become an indispensable tool in understanding public opinion in various domains. This project leverages the BERT (Bidirectional Encoder Representations from Transformers) model, a pre-eminent deep learning approach in Natural Language Processing (NLP), to accurately discern sentiment from textual reviews in the Yelp dataset. The model was fine-tuned and customized to suit the tri-class sentiment classification task, converting the 1-5 star ratings into negative, neutral, and positive sentiments. During training, the model demonstrated a promising ability to learn, achieving an accuracy of 82.00% on the validation set. The learning rate was managed with an exponential decay schedule, starting at 5e-5, to optimize the training process. Upon evaluation on a separate test set, the model sustained its performance, achieving a test accuracy of 83.06% with a corresponding loss of 0.4640. This report details the methodologies employed, presents the findings of the training and evaluation phases, and discusses the implications of the results. The high accuracy on the test set underscores the efficacy of fine-tuning BERT for sentiment analysis, opening avenues for future research to explore its application across diverse domains.

1 Introduction

Sentiment analysis, a subfield of Natural Language Processing (NLP), is concerned with the detection and interpretation of subjective information in text data. It is instrumental in various applications such as understanding consumer feedback, gauging public sentiment on social issues, and automating customer service responses. The motivation for this project stems from the need to discern the underlying sentiment in user-generated content, which has proliferated with the advent of social media and review platforms.

This project specifically focuses on analyzing sentiments expressed in Yelp reviews, a rich dataset of user opinions about various businesses and services. Yelp reviews are an invaluable resource for businesses and researchers alike, as they contain candid insights into consumer experiences. The dataset comprises textual reviews paired with star ratings, which serve as a proxy for sentiment. The project aims to classify these reviews into three sentiment categories: negative, neutral, and positive.

The cornerstone of the project is the BERT model, a transformative approach in NLP introduced by Google in 2018. BERT stands for Bidirectional Encoder Representations from Transformers and represents a paradigm shift in how contextual information is extracted from text. Unlike previous models that processed text in one direction (either left-to-right or right-to-left), BERT is designed to understand the full context of a word by looking at the words that come before and after it—hence the term 'bidirectional'. BERT's architecture is built upon the Transformer, a deep learning model that employs attention mechanisms to weigh the influence of different words in a sentence. Due to its deep contextualized nature, BERT has set new benchmarks in a wide array of NLP tasks.

In this project, a fine-tuned BERT model is employed to predict the sentiment of Yelp reviews. The model is trained on a preprocessed version of the Yelp reviews dataset, where text data is cleaned, tokenized, and encoded before being presented to the neural network. The process of fine-tuning allows the pre-trained BERT model to adapt to the nuances of the sentiment classification task at hand. This report documents the methodology followed, presents the results of the training and evaluation phases, and discusses the implications and potential of the implemented model.

2 Methodology

This section outlines the systematic approach taken to address the sentiment analysis task using the Yelp Reviews dataset. The methodology encompasses data preprocessing, model architecture design, and the training process.

2.1 Data Preprocessing

The raw Yelp Reviews dataset, containing textual reviews and corresponding star ratings, underwent several preprocessing steps to prepare it for the sentiment analysis task. The preprocessing pipeline was as follows:

- **Text Cleaning:** The reviews were stripped of punctuation and transformed to lowercase to standardize the text and reduce the complexity of the vocabulary.

- **Stopword Removal:** Common English stopwords were removed using the Natural Language Toolkit (NLTK) to focus on the most meaningful words in the reviews.

- **Sentiment Labeling:** Star ratings were mapped to sentiment labels with ratings above 3 categorized as 'Positive', equal to 3 as 'Neutral', and below 3 as 'Negative'.

- **Label Encoding:** The sentiment labels were then encoded into numerical format for model processing, with ‘Negative’ as 0, ‘Neutral’ as 1, and ‘Positive’ as 2.
- **Data Splitting:** The dataset was split into training, validation, and test sets using an 80-20 ratio for the training-validation split.

2.2 Model Architecture

The model architecture was built upon the BERT model, leveraging its pre-trained weights and bidirectional training mechanism. The architecture was customized as follows:

- **BERT Embedding:** Input textual data was tokenized and encoded using BERT’s tokenizer, and passed through the BERT model to obtain contextual embeddings.
- **Pooling Layer:** The BERT output was then pooled using a Global Average Pooling layer to reduce dimensionality and extract the most salient features.
- **Dense and Dropout Layers:** A dense layer with 64 units and ReLU activation was added for further learning, followed by a dropout layer with a rate of 0.5 for regularization.
- **Output Layer:** The final layer was a dense layer with a softmax activation to classify the reviews into the three sentiment categories.

2.3 Training

The training procedure for the sentiment analysis model was as follows:

- **Learning Rate Scheduling:** An exponential decay schedule was applied to the learning rate starting at 5e-5, decaying by a factor of 0.9 every 10,000 steps to fine-tune the learning process over time.
- **Optimization:** The Stochastic Gradient Descent (SGD) optimizer was utilized to update the network weights, with the model compiling using categorical cross-entropy as the loss function due to the categorical nature of the sentiment labels.
- **Checkpointing:** The ModelCheckpoint callback was employed to save the model with the highest validation accuracy, preserving the best-performing weights throughout training.
- **Training Loop:** The model was trained for 3 epochs, with performance on the validation set being monitored after each epoch. Training progress was visualized through accuracy and loss graphs for both the training and validation sets.

3 Results

The results of the sentiment analysis project are presented in two parts: the outcomes of the model training process and the performance evaluation on the test set.

3.1 Training Results

The model training was executed over 3 epochs, with the accuracy and loss metrics recorded for both the training and validation datasets. The training process was visualized using Matplotlib to plot the accuracy over epochs.

During the initial epoch, the model attained a training accuracy of 67.61% and a validation accuracy of 75.11%, with corresponding losses of 0.8097 and 0.6841, respectively. As the epochs progressed, the model showed improvement in learning, evidenced by an increase in accuracy and a decrease in loss for both the training and validation sets. By the final epoch, the model reached a training accuracy of 82.00% and a validation accuracy of 83.33%, demonstrating a consistent learning trend with the validation loss reducing to 0.4599.

The training curve (Figure 1) displays a steady increase in accuracy, indicating effective learning and generalization capability of the model without signs of overfitting, as the validation accuracy closely follows the training accuracy.

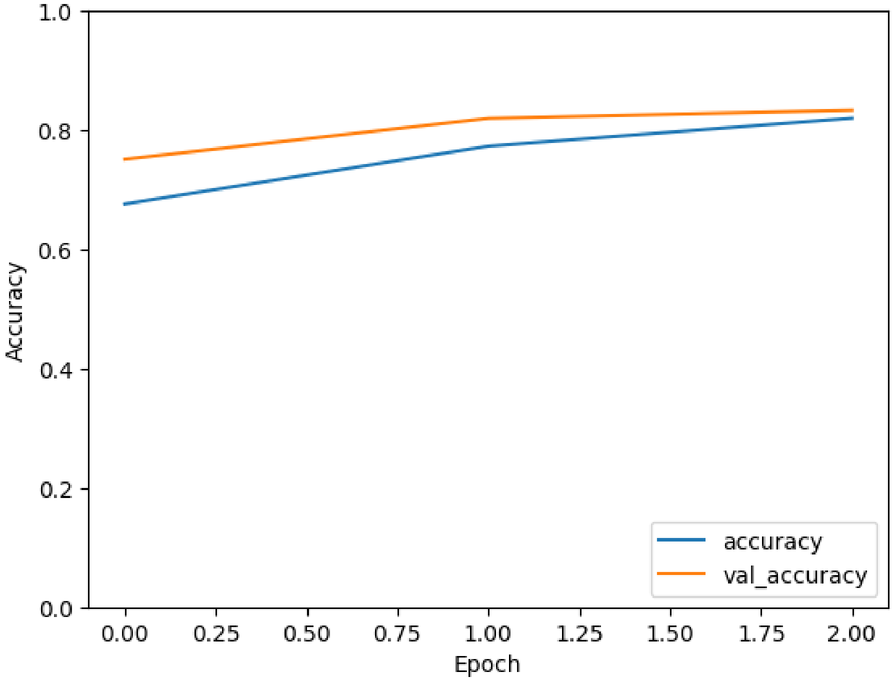

Figure 1: Training and validation accuracy over epochs.

3.2 Evaluation Results

The best model, as saved during the training process, was loaded and evaluated on the test dataset to gauge its performance on unseen data. The test set consisted of Yelp reviews that the model had not been exposed to during training. The evaluation yielded a test accuracy of 83.06% with a loss of 0.4640, corroborating the model's generalization ability as observed during the validation phase.

The test results indicate that the fine-tuned BERT model is adept at sentiment analysis, aligning with the performance seen during the validation phase and confirming the model's robustness.

4 Conclusion

This project aimed to apply the BERT model, a powerful neural network architecture for natural language understanding, to the task of sentiment analysis on Yelp review data. Through meticulous data preprocessing, careful model customization, and strategic training, the model achieved commendable performance on both the validation and unseen test datasets.

The findings indicate that the BERT model, even with a relatively small amount of fine-tuning, is highly effective for sentiment classification tasks. The training process demonstrated a strong learning capability, with validation and test accuracies exceeding 83%. These results underscore the adaptability of BERT to specific NLP tasks and its potential as a tool for analyzing and interpreting large volumes of textual data.

The implications of this project are significant for businesses and platforms that rely on user-generated content. By automating sentiment analysis, organizations can gain rapid insights into customer opinions and experiences, allowing for more responsive and data-driven decision-making.

For future work, there are several promising avenues to explore:

- Expanding the dataset to include a broader range of review sources could further improve the model's robustness and generalizability.
- Experimenting with different hyperparameters and training strategies, such as learning rate warm-up and more sophisticated optimizers, may yield further improvements in model performance.
- Investigating model interpretability techniques to understand the model's decision-making process could provide deeper insights into its predictions and help identify any potential biases.
- Deploying the model as part of a real-time sentiment analysis pipeline could test its effectiveness in a production environment and open up possibilities for real-time feedback analysis.

In conclusion, the project successfully demonstrates the utility of the BERT model for sentiment analysis and lays the groundwork for future enhancements and applications.