Report on Final Project

December 13, 2023

1 Abstract

Sentiment analysis is essential for gaining valuable insights from large volumes of unstructured textual data, especially when it comes to social media consumer reviews. This study presents a sophisticated sentiment analysis algorithm that makes use of Transformers' Bidirectional Encoder Representations (BERT). The cutting-edge language model BERT is renowned for its performance in a range of sequential modeling applications and natural language processing jobs.

This update seeks to fully capture semantic information down to the sentence level, improving the model's capacity to identify complex emotions in customer evaluations. In order to accomplish this, we provide a novel speech rule-based recognition technique that enables us to discern the emotional inclinations that consumers express.

2 Introduction

In the landscape of Natural Language Processing (NLP), the Bidirectional Encoder Representations from Transformers (BERT), introduced by Google in 2018 [1], has emerged as a groundbreaking and open-sourced language model. BERT operates on the principles of bidirectional text representation, pre-training on vast amounts of unlabeled text data to capture nuanced semantic information [2]. The original BERT models, namely BERTBASE and BERTLARGE, differ in the number of Encoders and bidirectional attention heads they incorporate.

BERT's pre-training involves exposure to 800 million words from BooksCorpus and 2.5 billion words from English Wikipedia's unlabeled text. This extensive pre-training allows BERT to be easily fine-tuned for specific NLP tasks, making it particularly suitable for sentiment analysis on small datasets derived from customer or employee reviews and question-answering systems for chatbot applications.

Sentiment Analysis (SA) stands as a critical domain within NLP, encompassing the classification of opinions and the mining of subjective texts with emotional tones [3]. The tasks associated with sentiment analysis span feature extraction, polarity classification, retrieval, and generalization [4]. Traditional machine learning and deep learning-based sentence classification methods have shown competence in short text classification; however, their static word vectors struggle to capture diverse word meanings in various contexts.

In recent years, BERT has emerged as a State-of-the-Art (SOTA) NLP model, utilizing pretraining tasks to extract semantic knowledge from large unlabeled corpora and enhancing semantic feature extraction [5]. Leveraging BERT's classification function, researchers have developed models for sentiment and complaint classifications based on energy-related tweets [6]. Additionally, ensemble learning methods have been combined with BERT for identifying harmful news, showcasing its versatility in various applications [7].

Despite the advancements, challenges persist in constructing comprehensive entity features in long sentences, particularly in the presence of asymmetric entity relationships. Moreover, sentiment analysis of Chinese text remains a formidable task due to the inherent lack of physical separation between Chinese words. This paper explores the potential of an enhanced BERT model [8], employing a symmetrical structure, to address these challenges and improve sentiment analysis accuracy in the context of agricultural product evaluations.

In this project yelp dataset was used for the analysis of the ratings given by the users and based on that sentiments are analysed. In the project, Transformer, BERT model was used.

3 Methodology

The following section will give the overview of the methods and the process that was used in the project:

3.1 Pandas

Pandas is a robust Python data manipulation and analysis package that has grown to be an important tool for data science and research. Pandas, an efficient and adaptable data structure called a DataFrame that is built on top of the Python programming language, makes it simple for researchers to handle, clean, and analyze tabular data. Because of its adaptability to many data kinds and formats, it is especially useful for tasks involving statistical analysis, exploratory data analysis, and data preprocessing. Pandas is frequently used by researchers to perform operations on datasets, including handling missing values and filtering, grouping, and aggregating data. Its ability to visualize and analyze research

findings is further enhanced by its smooth interface with other libraries, like NumPy and Matplotlib.

3.2 Numpy

NumPy is an open-source numerical computing library for Python that offers a strong set of mathematical functions to manipulate huge, multi-dimensional arrays and matrices, in addition to providing necessary functionalities for handling big arrays. Because of its adaptability and effectiveness, it is a vital tool for researchers in a variety of fields, enabling them to easily complete intricate mathematical operations, statistical analysis, and data manipulations. Because NumPy's underlying C and Fortran implementations guarantee computational efficiency, researchers may analyze large datasets and run complex algorithms as efficiently as possible. Its easy interaction with other scientific frameworks and libraries increases its usefulness even more and encourages an interdisciplinary and collaborative approach to study.

3.3 Matplotlib

Matplotlib is a cornerstone in the field of data visualization, improving the graphical depiction of research findings. With the help of this extensive and adaptable plotting library, researchers may create a wide variety of static, animated, and interactive visualizations with ease. Matplotlib is a scientific computing library designed to work seamlessly with NumPy and Pandas. It enables academics to create visually appealing and publication-ready plots, charts, and graphs. Its adaptability to many visualization styles—such as line plots, scatter plots, histograms, and heatmaps—meets the needs of researchers in a variety of fields.

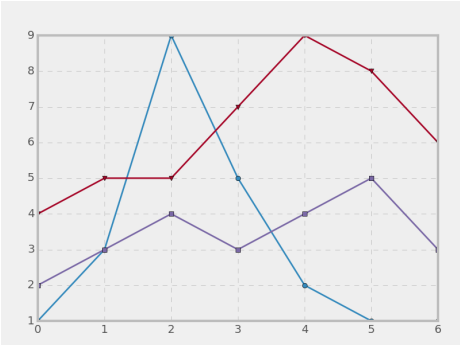

Figure 1: Matplotlib

In Figure 1, shows an example of graph which is drawn using matplotlib library.

3.4 Data Preprocessing

Data preprocessing is a crucial step in natural language processing tasks, ensuring that raw text data is transformed into a format suitable for analysis. The Natural Language Toolkit (NLTK) in Python provides a comprehensive set of tools for text processing and analysis.

3.5 Tokenization

Tokenization is the process of breaking down a text into individual words or phrases, known as tokens. This step is essential for further analysis.

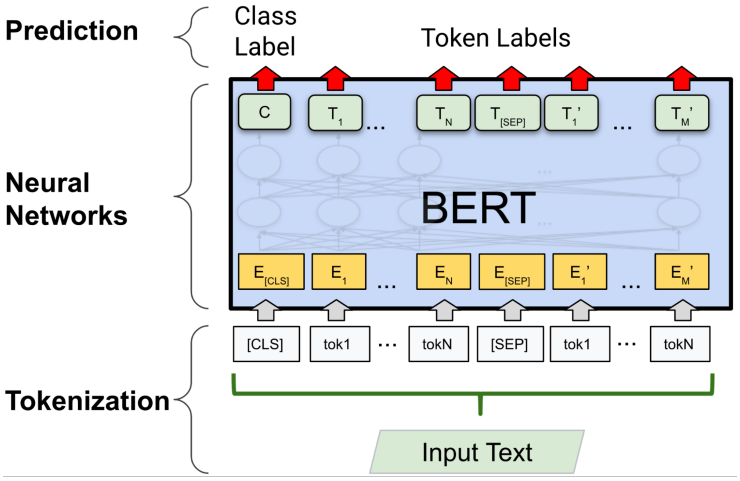

Figure 2: Process of tokenization

In Figure 2, the tokenization process involves breaking down a sentence into individual tokens.

3.6 Stopword Removal

Stopwords are common words (e.g., "and," "the," "is") that often do not contribute much information to the analysis. Removing stopwords can help reduce noise in the data

Data preprocessing using NLTK is a fundamental step in preparing text data for analysis. The tokenization, stopword removal are essential for cleaning and transforming raw text into a more structured and meaningful format.

3.7 Transformers

Transformers are a revolutionary architecture in machine learning that transform the modeling and processing of sequential data. Transformers were first presented by Vaswani et al. in the paper "Attention is All You Need," and since then, they have dominated the field of natural language processing (NLP) and other related fields. Transformers, as opposed to conventional recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, rely on a self-attention mechanism, which makes it easier for them to identify complex dependencies in sequential data.

The encoder-decoder structure of the transformer architecture has numerous attention heads in each layer. Attention mechanisms enable the accumulation of information across various sequence places, while the encoder processes input sequences and the decoder creates output sequences. Large text corpora can be used to pre-train transformers, which help them acquire rich contextual embeddings that can be tailored for certain downstream tasks.

Transformers are useful in a variety of fields outside of natural language processing (NLP), such as computer vision, audio processing, and reinforcement learning. Pre-trained transformer models have demonstrated the versatility and effectiveness of this architecture by setting new records in a variety of tasks. Examples of these models are BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer).

Figure 3: Transformer for sentimental analysis

3.8 BERT

BERT is pre-trained on massive amounts of unlabeled text data, such as the BooksCorpus and English Wikipedia. The model learns to predict missing words in a sentence by considering the surrounding words bidirectionally. The result is a set of rich, context-aware embeddings that can be fine-tuned for various downstream NLP tasks, including sentiment analysis, named entity recognition, and question answering.

One notable feature of BERT is its attention mechanism, which allows the model to focus on different parts of the input sequence when making predictions. This attention to context is crucial for understanding the semantics of natural language and has contributed to BERT's exceptional performance on a wide

range of benchmarks.

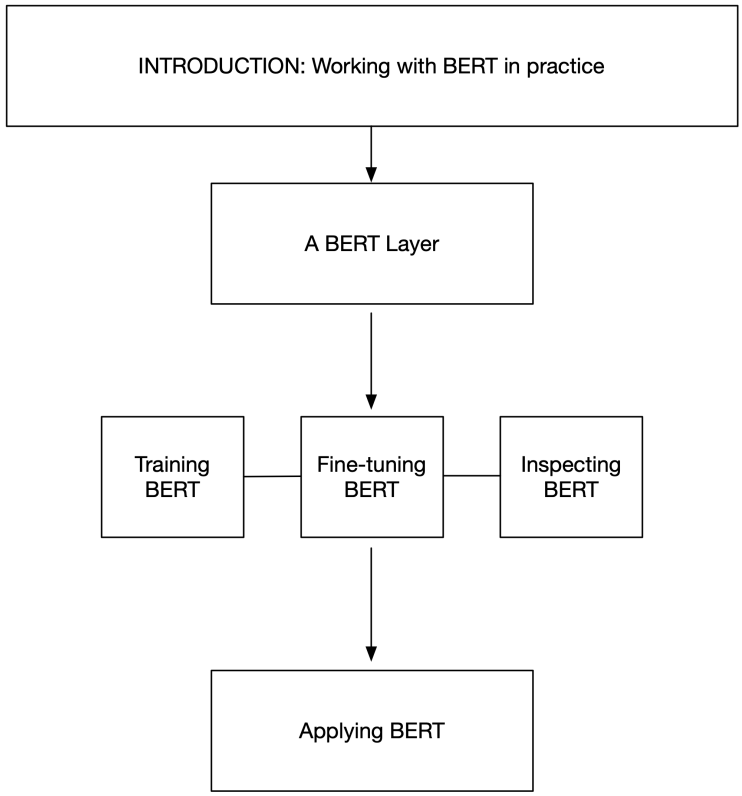

Figure 4: BERT

3.9 Why BERT?

Context-Aware Representations:
BERT, being bidirectional, captures contextual information from both left and right contexts in a sentence. This is beneficial for tasks like sentiment analysis where understanding the overall context of a statement is crucial.

Fine-Tuning:
BERT can be easily fine-tuned on specific sentiment analysis tasks using a relatively small amount of labeled data. This makes it suitable for domain-specific sentiment analysis applications.

4 Experiments

This section talks about the experiments, flow of the code and the setting of the parameters.

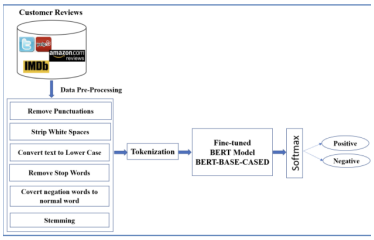

Figure 5: Implementation

4.1 Implementation

1. Import the libraries
Import all the required libraries, load the dataset and take a look of it using df.head().
Libraries used are -
import transformers from transformers import BertModel,
BertTokenizer, AdamW,
get-linear-schedule-with-warmup
import torch
import torch.nn as nn
import numpy as np
import seaborn as sns
import pandas as pd
import nltk
import re
from nltk.corpus import stopwords
import matplotlib.pyplot as plt
from sklearn.model-selection
import train-test-split

2. preprocessing For data preprocessing, downloaded the NLTK stopwords and then defined a text preprocessing function under which it applies the preprocessing to a 'text' column in a DataFrame, removing stopwords and converting text to lowercase and then it categorizes sentiment based on star ratings into negative, neutral, and positive based on following conditions, Positive $>$ 3, negative $<=$ 2, and neutral $=$ 3 levels.

3. Imbalance data After preprocessing, check if the data is balanced or imbalanced by using .count() and the result come out that the data was balanced.
If the data was imbalanced then the imbalance in a dataset affects the optimal performance of the model.
After the check of balance of data, stratisfied sampling to done to ensure that the class distribution is maintained.
After sampling we merge the training and validation DataFrames into a single DataFrame because the target distribution is same.

4. Tokenization First upload the pre trained model -'bert-base-cased' and after that we'll use pre-trained tokenizer to evaluate the density of the data.

Experimented with the max-len of the sentence. The max length of the sentence is 520 but the system was getting crashed so to make it work, we experimented with different lenghts in decreasing order and 512 length worked well.

5. Creating Pytorch dataset loader We will define a custom PyTorch dataset ('ReviewDataset') for text reviews and corresponding sentiment targets. The dataset utilizes a tokenizer to encode the text, ensuring a specified maximum length. It will return a dictionary containing the original review text, input IDs, attention mask, and the sentiment target, suitable for training a neural network. Batch size is a crucial aspect of training machine learning models, including those used in sentiment analysis. The batch size determines the number of samples that will be processed in each iteration during training.

So, Experimented with the batch - size of the data. First took 64 as size but it didn't work as it was taking alot of memory and then tried with the half of the number that previously took which is 32 and it worked very well.

6. Creating BERT model We will create a BERT model and train it with hyperparameter as number of epochs and is equal to 10 and then train and evaluate the model on each epochs but later when we evaluate the model it results into overusage of RAM. So slowly decreased the number of epochs and it reaches to 2 when it started working again and give the result in terms of loss and accuracy of training and validation set respectively.

7. Evaluation
After creating the BERT model and training the model on each epoch, accuracy is tested on the test set. Further we will perform the predictions on provided test set and will compute the results and decide if the model prediction is good enough or not.

5 Results

In this experiment, Precision (P), Recall (R), Accuracy and F1 values were used as evaluation indexes. Precision (P) refers to the proportion of the number of correctly classified sentences to the number of all sentences predicted, and Recall (R) refers to the proportion of the number of correctly classified sentences to the number of true sentences for a particular classification. In practice, Precision (P) and Recall (P) are contradictory measures, where a high recall rate may result in a low accuracy rate, and vice versa. The F1 value is a combination of precision and recall, and the higher the F1 value, the better the model performance.

The following Table 1 shows that the accuracy of the model is 84 percent which implies that the model is working fine and can be used for the predictions.

	Precision	Recall	F1-Score
Negative	0.73	0.89	0.80
Positive	0.89	0.95	0.92
Neutral	0.46	0.06	0.11
Accuracy			0.84
Macro Avg	0.69	0.63	0.61
Weighted Avg	0.81	0.84	0.81

Table 1: Result

After getting the results we have implemented it on some text to check if it works correctly or not.
Example(that i gace to check the prediction) - I gave review text as
review-text = "the food was bad and I pucked after eating the food"
When i ran the code, it resulted into negative prediction which is true. This is another way which brings confidence into the result.

6 Conclusion

In this paper, we implemented a transformer BERT-BASE-CASED model for the sentimental analysis and the result comes out be good enough to say that the model works fine and can be used to experiment on other dataset.
In our future research endeavors, our objective is to employ this innovative approach across various crucial domains, verifying its applicability, and assessing the model's effectiveness on multilingual datasets to evaluate its performance in diverse languages.

References

[1] Jacob Devlin and Ming-Wei Chnag, *Research Scientists Google AI Lanaguage*, 2018.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, *BERT:Pre-training of Deep Birirectional Transformers for Lanaguage Understanding*, 2019.

[3] Deng.L, *Deep Learning: Methods and Applications. Found. Trends Signal Process*, vol. 7, pp. 197–387, 2014.

[4] Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. *Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 1–6, 2018.

[5] Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2018.

[6] Bedi, J.; Toshniwal, D. CitEnergy *A BERT based model to analyse Citizens' Energy-Tweets*, 2022.

[7] Lin, S.-Y.; Kung, Y.-C.; Leu, F.-Y *Predictive intelligence in harmful news identification by BERT-based ensemble learning model with text sentiment analysis*, 2022.

[8] Durairaj Ashok, Chinnalagu Anandan *Transformer based Contextual Model for Sentiment Analysis of Customer Reviews: A Fine-tuned BERT*, vol. 12, 2021.