Sentiment Analysis of Restaurant Reviews

Faranak Abri[1], Mrunal Deepak Zambre[1], Monica Meduri[2], Avinash Mangalore Suresh[3]
[1]Computer Science Department, [2]College of Science, [1,2]San Jose State University
{*faranak.abri, mrunaldeepak.zambre, monica.meduri, avinash.mangaloresuresh*}*@sjsu.edu*

***Abstract*—Reviews are key in determining the reputation of the restaurant and would reflect in the sales achieved by them. Customers heavily rely on restaurant reviews when deciding where to eat since they have so many options. Studies show that 90 percent of customers consider reviews before visiting an establishment. These reviews can be classified as positive/negative based on the way they describe the customer's experience. This project uses analysis techniques that analyze text that automatically detects the polarity of the reviews. Sentimental Analysis has gained much attention in recent years. In this project, we aim to categorize the reviews based on their polarity.**

***Index Terms*—Sentiment analysis; Sentiment polarity categorization; Natural language processing; Restaurant reviews**

[1]

I. INTRODUCTION

Sentiment analysis also known as opinion mining refers to the use of natural language processing, text analysis, and computational linguistics to identify and extract subjective information in the source materials. In other words, sentimental analysis is a type of natural language processing for tracking the mood of the public about a particular topic. This project takes reviews as input and assigns a sentiment as positive or negative. It involves building a system to examine reviews to classify them. The pre-trained models we used for this are BERT, RoBERTa, LUKE, XLNet and GPT.

II. RELATED WORK

Techniques for sentiment analysis have been developing in recent years. In general, sentiment analysis includes both the processes of feature extraction and sentiment classification. A typical feature extraction process includes data preprocessing, TF-IDF, and selection methods such as Odds Ratio and Chi-Square. The sentiment classification methods mainly include two approaches: lexicon-based, and machine-learning based, where support vector machines, neural networks, and trainable Bayesian networks are involved.

III. DATASET DESCRIPTION

The restaurant review dataset has 260 rows and 5 columns namely Restaurant, Review, Real=1/Fake=0, Positive=1/Negative=0, and Author before feature extraction. We have only considered the "Review" and "Positive=1/Negative=0" columns for the sentiment analysis task. We have renamed the "Positive=1/Negative=0" column to "labels" for easier readability.

[1]https://www.overleaf.com/read/nbhjmzbsnczs

978-1-7281-6251-5/20/$31.00 ©2020 IEEE

IV. FEATURES

The features used here include the reviews as plain text and the label of the respective review, indicating whether they are positive or negative.

V. TEXT PREPROCESSING

As part of this project we are operating under the assumption that model does not require any additional preprocessing, and can learn from raw data.

VI. CLASSIFICATION MODELS

Pre-trained models have been made available to support customers who need to perform tasks such as sentiment analysis or image featurization, but do not have the resources to obtain the large datasets or train a complex model. We have used 5 pre-trained models from the huggingface library, and fine-tuned them on our dataset.

A. BERT

A machine learning (ML) framework for dealing with natural language is called BERT. Bidirectional Encoder Representations from Transformers is another name for BERT. Google created this algorithm in 2018 to enhance contextual comprehension of unlabeled text across a variety of activities. It has the ability to anticipate text that might come both before and after (bi-directional) other text.

B. RoBERTa

RoBERTa builds on BERT's language masking technique, which trains the computer to identify deliberately concealed content inside samples of unannotated language. With the BERT modification RoBERTa, which was implemented in PyTorch, the next-sentence pretraining target is eliminated, and training is carried out with noticeably bigger mini-batches and learning rates. As a consequence, RoBERTa outperforms BERT in terms of the objective of masked language modeling and performs better on upcoming tasks.

C. LUKE

LUKE (Language Understanding with Knowledge-based Embeddings) is a new pre-trained contextualized representation of words and entities based on transformer. LUKE treats words and entities in a given text as independent tokens, and outputs contextualized representations of them. LUKE adopts an entity-aware self-attention mechanism that is an extension of the self-attention mechanism of the transformer, and considers the types of tokens (words or entities) when

computing attention scores. The LUKE base model has 12 hidden layers, 768 hidden size. The total number of parameters in this model is 253M. It is trained using December 2018 version of Wikipedia.

D. XLNet

XLNet model pre-trained on English language. It was introduced in the paper XLNet: Generalized Autoregressive Pretraining for Language Understanding by Yang et al. XLNet is a new unsupervised language representation learning method based on a novel generalized permutation language modeling objective. Additionally, XLNet employs Transformer-XL as the backbone model, exhibiting excellent performance for language tasks involving long context. Overall, XLNet achieves state-of-the-art (SOTA) results on various downstream language tasks including question answering, natural language inference, sentiment analysis, and document ranking. This model is primarily aimed at being fine-tuned on tasks that use the whole sentence (potentially masked) to make decisions, such as sequence classification, token classification or question answering.

E. GPT

GPT-2 is a transformers model pretrained on a very large corpus of English data in a self-supervised fashion. This means it was pretrained on the raw texts only, with no humans labelling them in any way (which is why it can use lots of publicly available data) with an automatic process to generate inputs and labels from those texts. More precisely, it was trained to guess the next word in sentences. This way, the model learns an inner representation of the English language that can then be used to extract features useful for downstream tasks. The model is best at what it was pretrained for however, which is generating texts from a prompt.

VII. Implementation

The implementation was done using Python scripting language as it has reliable support for data manipulation and machine learning tasks. Matplotlib library was used for visualizations of the results. For machine learning models, Scikit-learn library was used for dataset splitting. The pretrained models are available in the HuggingFace library. The Transformers and Datasets libraries were used for fine-tuning the pretrained models.

VIII. Results

To compare the performance and results of the five models, we have used 2 evaluation metrics - Accuracy and F1 score. Accuracy is a metric for evaluating classification models. It is the ratio of number of correct predictions to the total number of input samples. It is given by the formula -

$$Accuracy = \frac{Numberofcorrectpredictions}{Totalnumberofpredictionsmade}$$

F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1]. It tells you how precise your classifier is (how many instances it classifies correctly), as well as how robust it is (it does not miss a significant number of instances).

$$F1score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

TABLE I: Results

Models	Accuracy	F1 score
RoBERTa	0.94	0.94
BERT	0.92	0.93
LUKE	0.98	0.98
XLNet	0.98	0.98
GPT	0.75	0.75

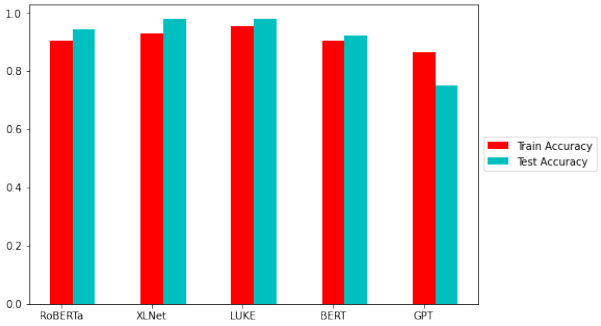

Fig. 1: Comparison of Accuracies

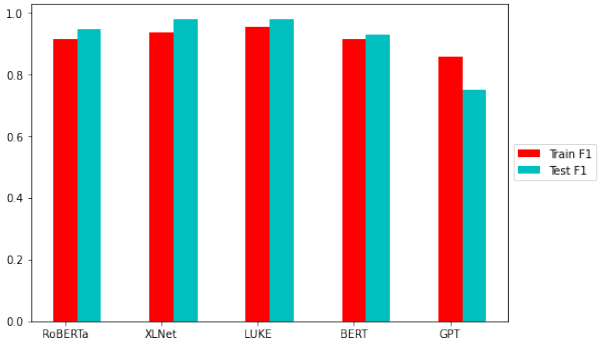

Fig. 2: Comparison of F1 scores

We can see the results in Table 1.We do not see a drastic difference in the performance of most models, which is in line with the fact that all these models are closely related or derivatives of each other, following a common underlying principle. Also the relative simplicity of the dataset contributes to the similar performance. We do see that GPT-2 performs the worst in comparison. Among the best models are XLnet and LUKE, having marginally higher accuracy and F1 scores.

IX. Conclusion

From the above results, it can be concluded that the best pre-trained models were LUKE and XLnet, with accuracy and f1 scores as 0.98 each. The next best models are RoBERTa, BERT, and GPT respectively.

ACKNOWLEDGEMENT

We'd like to humbly acknowledge Professor Faranak Abri. This project would not have been possible without her constant guidance and support.