# COMP 4332 Project 1 Sentimental Analysis

Group 33: Lee Jihyun, Park Juhyeon, Son Moo Hyun

***Abstract*—With the rise of social media and online reviews, it has grown in popularity. In this project, we concentrate on categorizing the sentiment of customer reviews posted online, which can offer businesses insightful data.**

## I. Introduction

The objective of this project is to create a sentiment classification model that can accurately predict the sentiment of business reviews. The model will categorize the reviews into 5 ratings from 1 to 5. This classification will help business owners and customers make informed decisions when reading reviews in various domains.

## II. Data Collection and Preparation

### A. Data Cleaning

Prior to exploration of data, first clean the dataset for a better performance. The cleaning process includes removing links, punctuation, digits, and expanding contractions. To maintain the negative context of the text, a negation cue was added by combining ”not” with an underscore, resulting in ”not_”. This step was necessary as ”not” is a highly frequent word that could potentially affect the accuracy of sentiment analysis.

### B. Tokenization

After cleaning the text, tokenize the text using the ‘spacy’ library. We set stopwords of the tokenizer as default setting and manually including the words ‘cannot’ and ‘not’. In the tokenizing process punctuations, stop words, and pronouns are removed once again.

### C. Lemmatization

In order to catch the identical words in different formats, we lemmatized the tokens before vectorizing them. This process is included in the tokenizer method.

### D. Vectorization

Various vectorization methods were performed and studied: Countervectorizer, word2vec, TF-IDF vectorizer, and Glove embedding-matrix. The vectorizing method is different from model to model.

## III. Data Analysis

### A. Distribution

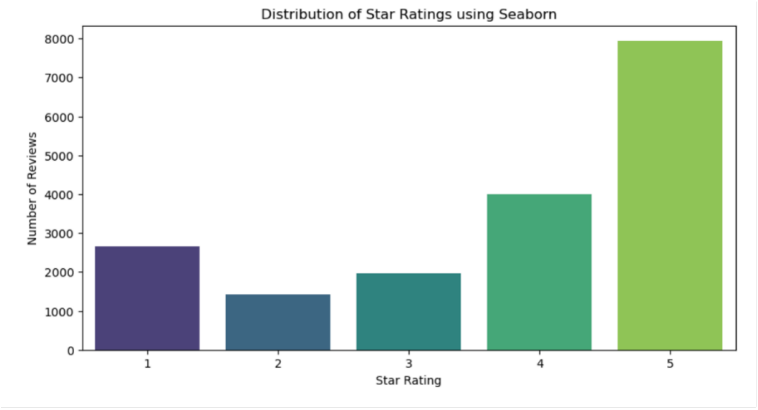


Fig. 1. Rating Distribution

### B. Word Frequency

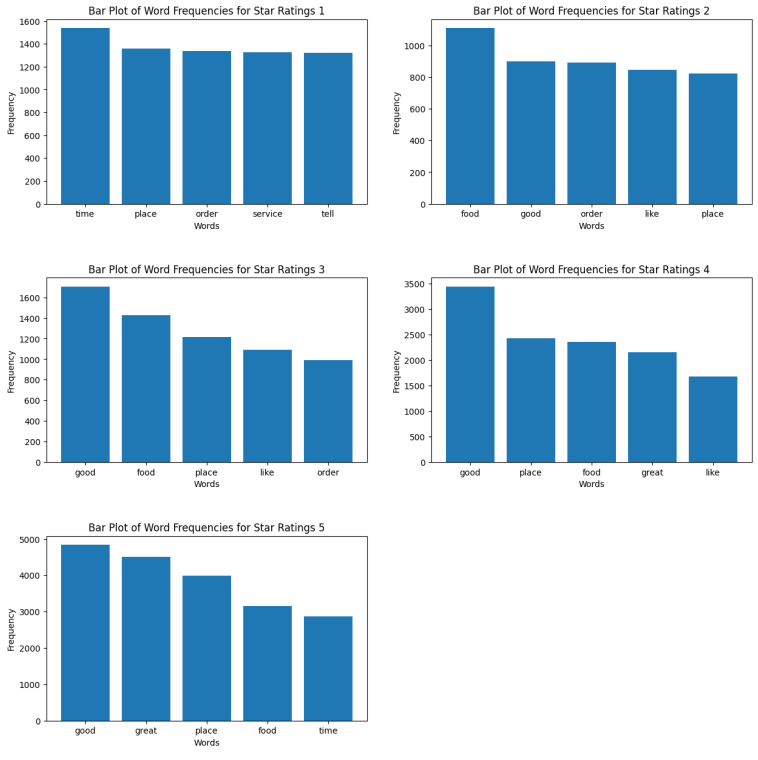


Fig. 2. Bar Plot of Word Frequency

### C. Word Cloud


Fig. 3. Word Cloud for Entire Data

### D. Distribution of Review Length

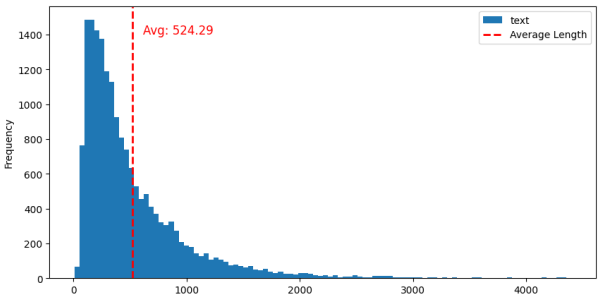


Fig. 4. Length of Texts

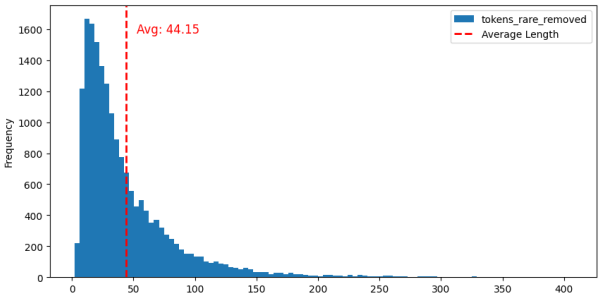


Fig. 5. Length of Tokens

## IV. Model Selection and Evaluation

### A. Conventional Machine Learning

Combinations of CountVectorizer, TF-IDF + Logistic Regression, Decision Tree Classifier, Random Forest Classifier, CatBooost, Gradient Boosting, multinomial Naïve Bayes, Kneighbors, support vector classifiers were tested. TF-IDF + CatBoost performed the best for each baseline model. TF-IDF + CatBoost was further studied using GridSearchCV for hyperparameter tunning.

### B. Deep Learning

Combinations of word2vec, Glove embedding + BiLSTM, BiGRU were tested. word2vec + BiLSTM and Glove embedding + BiGRU were further studied due to their high performance.

Analyzing the classification report, it is evident that the f1-scores for ratings 2, 3, and 4 are lower than those for ratings 1 and 5. This discrepancy can be attributed to the imbalance present within the dataset. To mitigate this issue, several approaches were implemented, including oversampling minority classes, assigning different class weights, and decomposing the multi-class problem into a series of binary classifications.

### C. Pre-trained Model

BERT (Bidirectional Encoder Representations from Transformers) + BiGRU (# of RNN layer = 1, # of MLP layer = 1) showed the best performance. The high performance is the result of transfer learning, a technique that leverages a pre-trained model trained with extensive data and a large number of parameters. By fine-tuning the model and employing its built-in tokenizer, the overall performance significantly enhanced.

TABLE I
Model Performance Comparison

| Models | F1 Score | Accuracy |
|---|---|---|
| Strong Baseline | 0.54 | 0.64 |
| TF-IDF + Catboost | 0.59 | 0.66 |
| word2vec+BiLSTM | 053 | 0.63 |
| Pre-trained BERT+BiGRU | 0.61 | 0.69 |

## V. Results and Discussion

While some models like TF-IDF applied with Catboost exceeds the strong baseline of validation performance, the BiGRU with BERT shows the best performance among the models studied. Other models also exceed the weak baseline but are not enough to satisfy the strong baseline. During the project, we identified various limitations.

Our predictive modeling might be biased towards the majority classes of rating 1 and 5 where 2, 3, and 4 are considered minority classes. To solve this problem, we tried to apply random oversampling method to the minority classes; however, this lowered the model performance due to the overfitting issue, therefore we excluded this unnecessary process in the code notebook. The data is insufficient to obtain a more precise result. If the dataset were larger, it might be better to train the models. The inclusion of reviews from multiple domains such as restaurants and hotels in the dataset used for analysis could potentially lower the accuracy of predictions, as the choice of words used in reviews may differ depending on the type of service being reviewed.

## VI. Conclusion

This project successfully developed a sentiment analysis model using NLP techniques for classifying reviews. The model can be integrated into platforms to help businesses better understand customer opinions and make data-driven decisions.