# Assignment 2: Semantic Analysis via Text Classification

CS525: Natural Language Processing

Thabsheer Jafer Machingal

## **Dataset**

For assignment 2, we’re working on Amazon fine food reviews dataset[1]. The dataset has data collected over 10 years and has about 500,000 reviews on Amazon products. Reviews are ranked on a scale of 1 to 5, where 5 being a positive review.

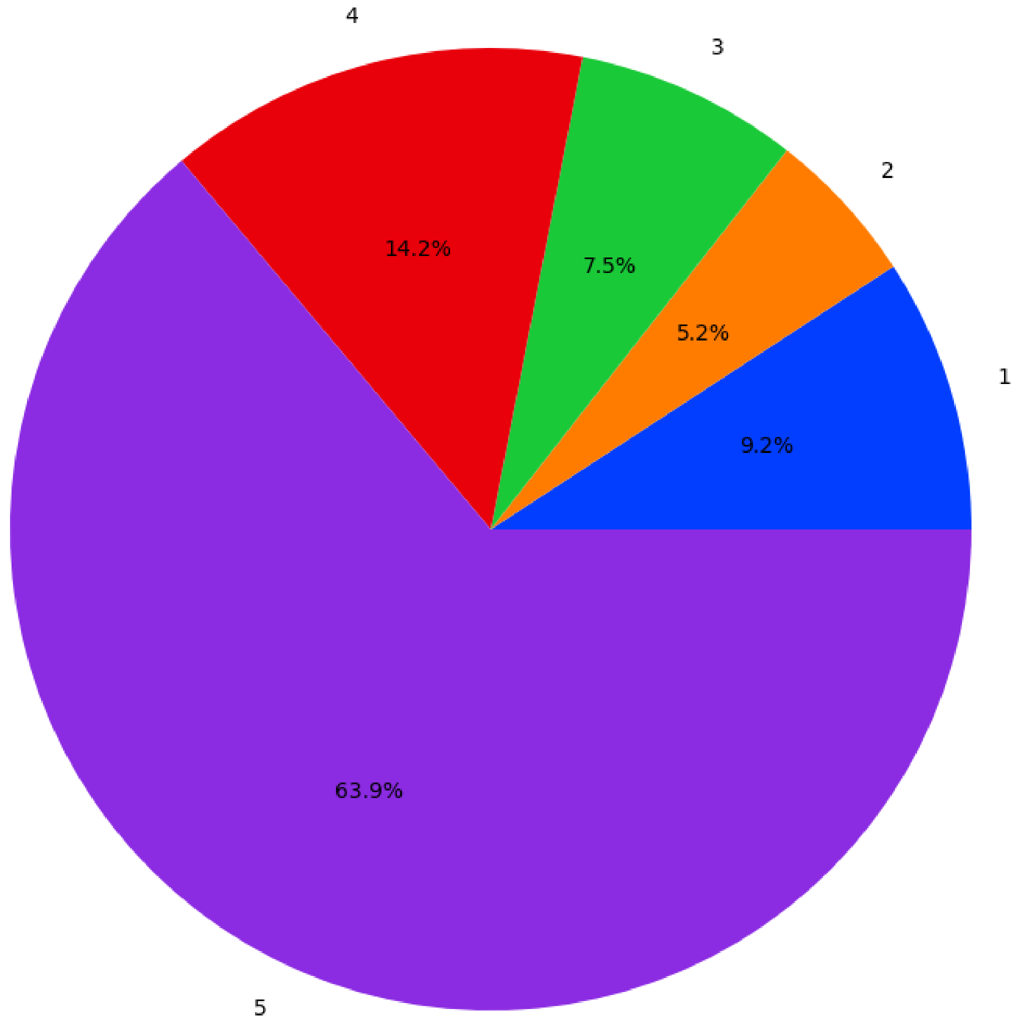


**Chart 1: Distribution of reviews on a scale of 1 to 5**

For this assignment, I have modified the class to either ‘Positive’ or ‘Negative’.where rating 1 and 2 is ‘Negative’ and 4 and 5 is labeled as ‘Positive’, I rejected the neutral reviews.
A fair amount of EDA was performed on the dataset to understand and each model and features requires tailoring the data as needed. For instance, I balanced the dataset among Positive and Negative reviews for word2vec model as it is desired, because the dataset is very unbalanced, one class is four times larger than the other (Chart 2).

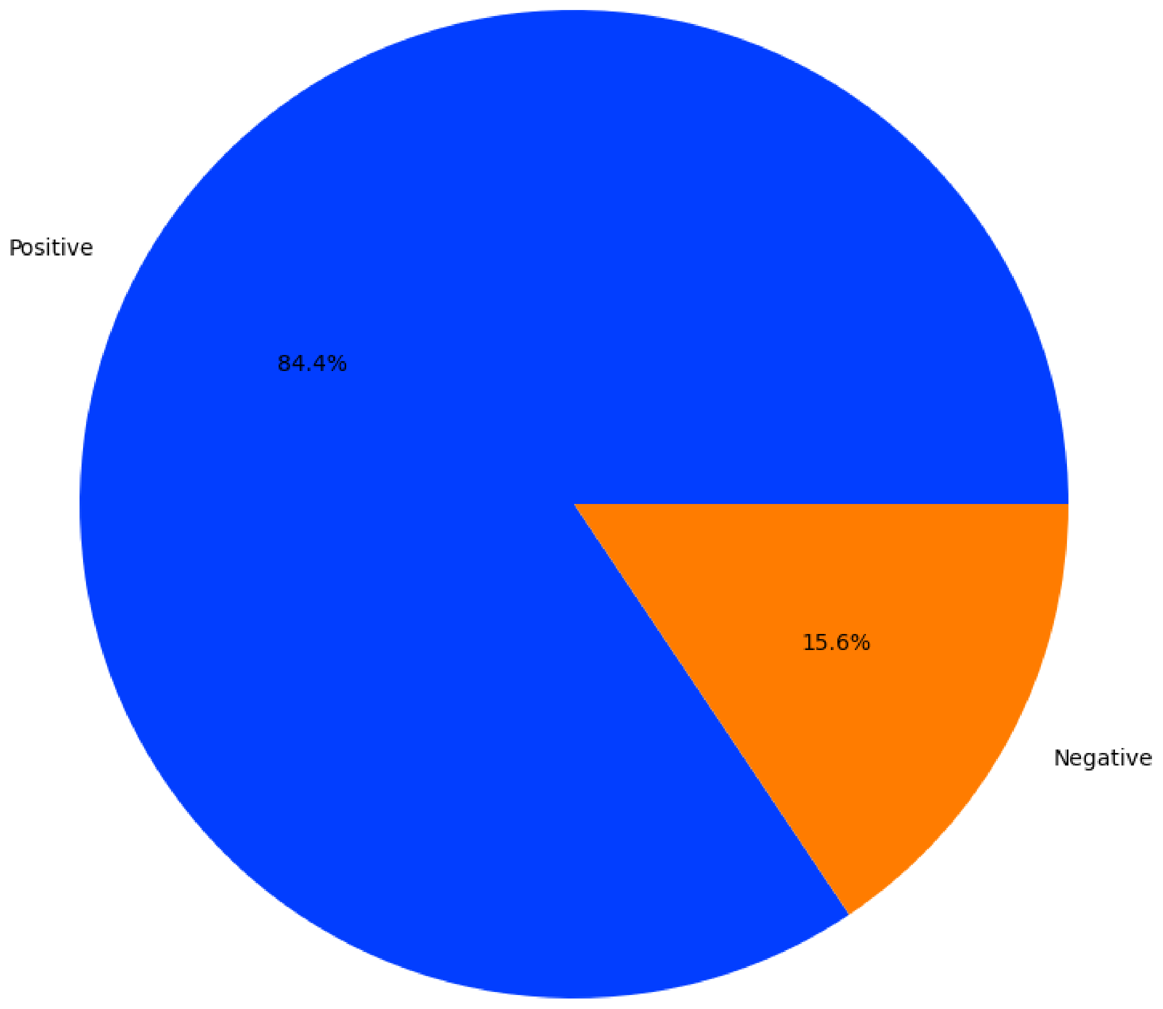


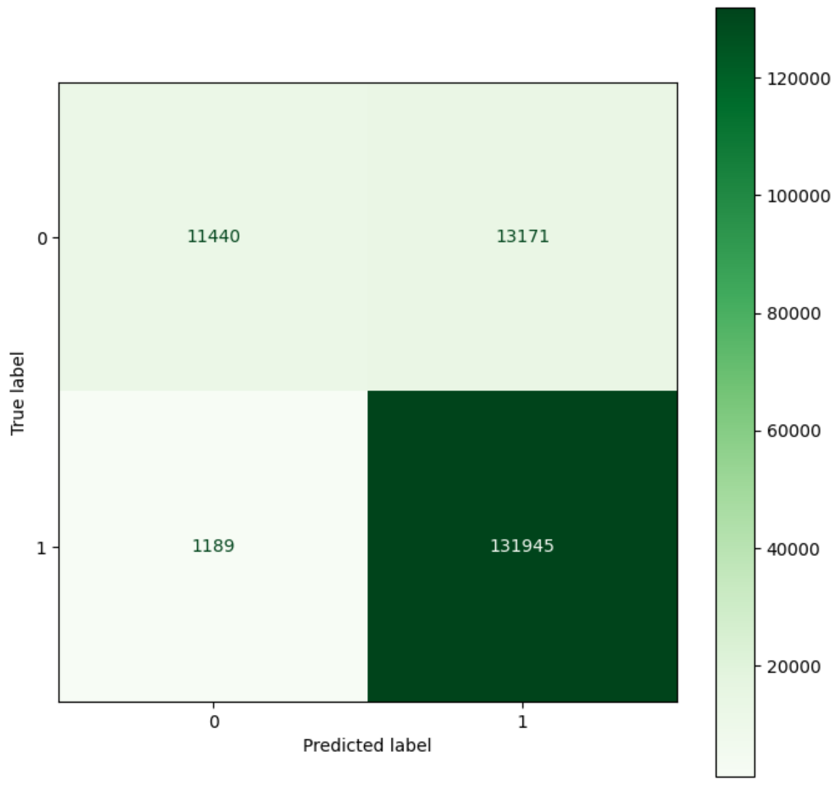
**Chart 2: Distribution of reviews among ‘Positive’ and ‘Negative’ reviews**

## Task 1 - Re-do TFIDF approach for this task

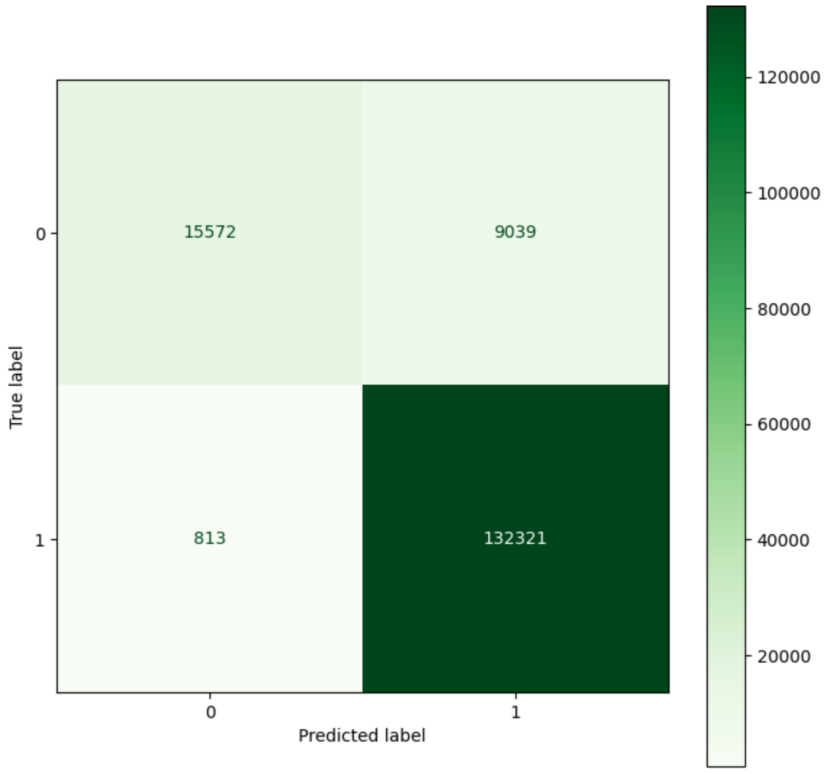
| ML Model | Feature | Precision | Recall | Accuracy | F1-score |
|---|---|---|---|---|---|
| Linear SVM | TF-IDF | 0.91 | 0.91 | 0.90 | 0.90 |
| Random Forest | TF-IDF | 0.94 | 0.94 | 0.93 | 0.93 |
| Multinomial Naive Bayes | TF-IDF | 0.88 | 0.88 | 0.87 | 0.85 |
| Perceptron | TF-IDF | 0.90 | 0.90 | 0.89 | 0.90 |

**Table 1: Table showing the performance on test set using TF-IDF**

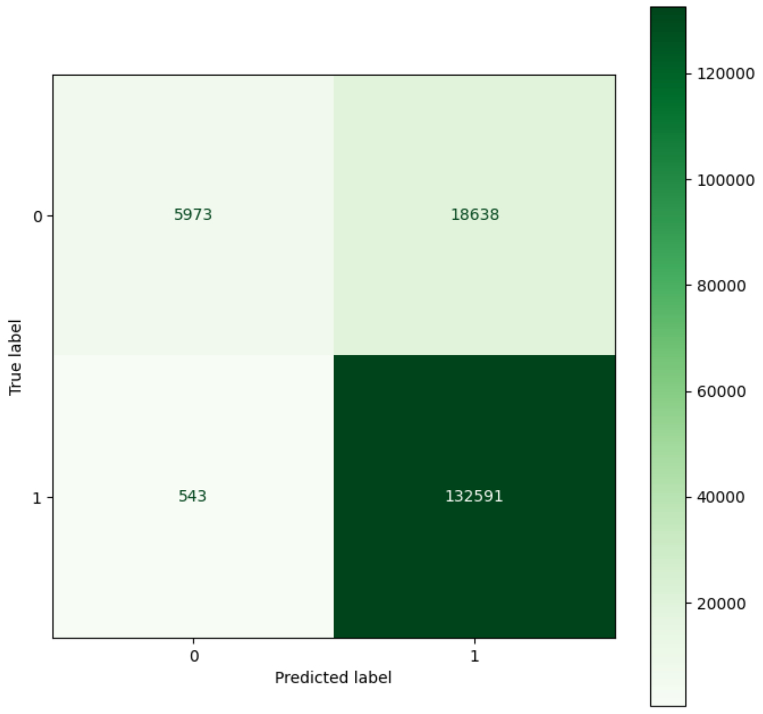
### CONFUSION MATRIX - Linear SVM



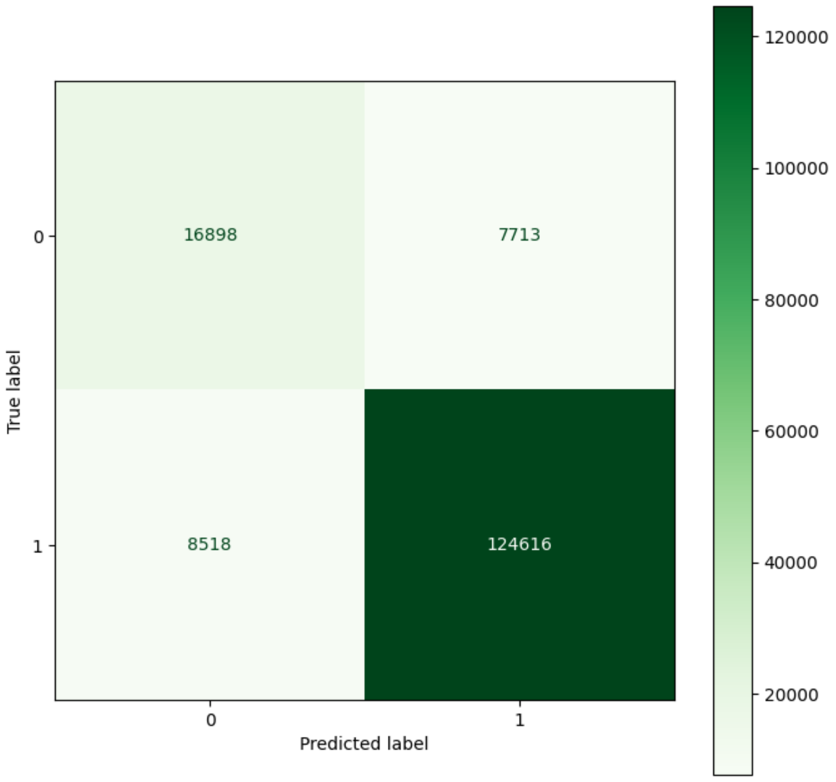
## CONFUSION MATRIX - Random Forest classifier



## CONFUSION MATRIX - Multinomial Naive Bayes



## CONFUSION MATRIX - Perceptron



## Task 2 -Review classification by using word2vec

| ML Model | Feature | Precision | Recall | Accuracy | F1-score |
|---|---|---|---|---|---|
| Linear SVM | word2vec | 0.72 | 0.69 | 0.68 | 0.67 |
| Random Forest | word2vec | 0.83 | 0.83 | 0.83 | 0.83 |
| Multinomial Naive Bayes | word2vec | 0.73 | 0.73 | 0.72 | 0.73 |
| Perceptron | word2vec | 0.71 | 0.70 | 0.70 | 0.70 |

**Table 2: Table showing the performance on test set using word2vec**

## Task 3 - BERT (without fine-tune) for review classification

For this task, a pretrained model from “huggingface.com” [2] is used. The input to the model is raw text and it gives out the label. The following results were observed using the model.

| ML Model | Precision | Recall | Accuracy | F1-score |
|---|---|---|---|---|
| BERT without fine-tuning | 0.74 | 0.74 | 0.74 | 0.74 |

**Table 3: Table showing the performance of BERT model without fine-tuning**

## Task 4 - BERT (with fine-tune) for review classification

| ML Model | Precision | Recall | Accuracy | F1-score |
|---|---|---|---|---|
| BERT when fine-tuned | 0.94 | 0.94 | 0.93 | 0.94 |

**Table 4: Table showing the performance of BERT model with fine-tuning**

## Task 5 - Results Analysis

The results from each task are presented above (Table 1 through 4).
TF-IDF showed better results compared to the word2vec model. In this assignment, I balanced the dataset for training with word2vec, this significantly reduced the dataset size, this could be one reason for lower accuracy. But the precision and F1-score is also low compared to that of models with TF-IDF.

## Word2vec vs TF-IDF

Tf-idf has better accuracy using all the tested models(see Chart3).

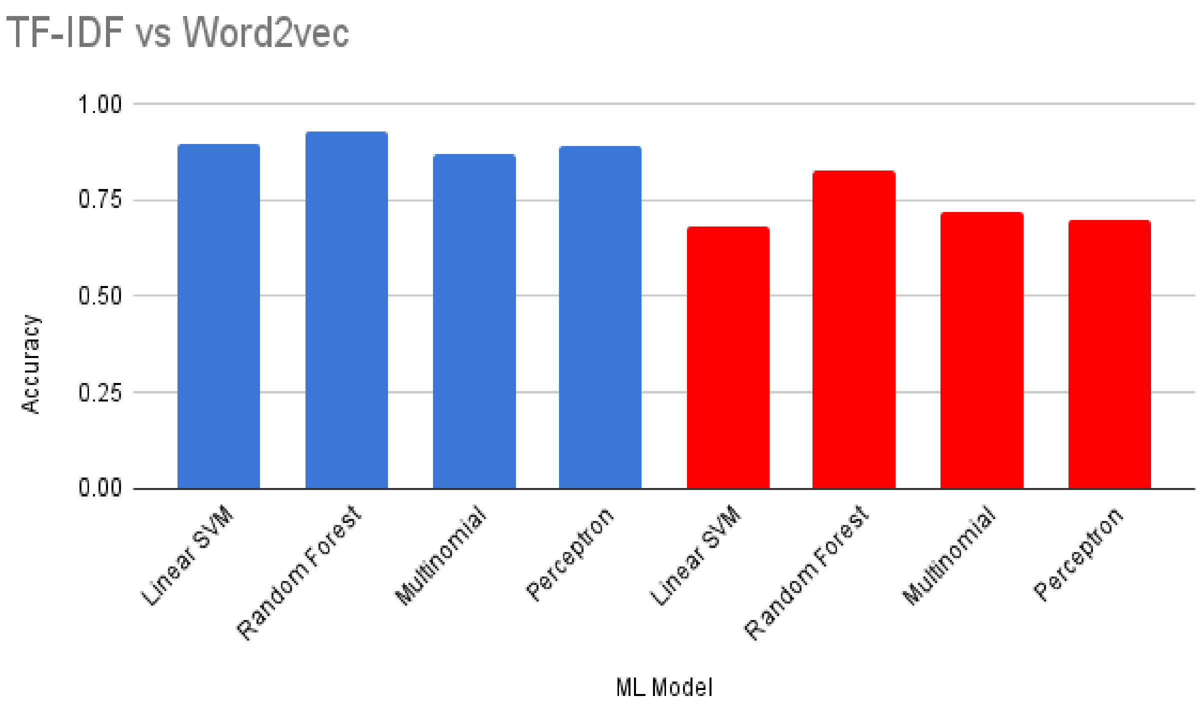


**Chart 3:Accuracy of different models using TF-IDF and Word2vec.**

## Word2vec

Since, Word2vec model finds one unique embedding for each words, it cannot generalize the words outside of the vocabulary. I trained the word2vec model on lesser texts to balance the dataset. This might have caused a problem if there is a word in the test set that was not on the train set.

## BERT (with and without fine-tuning)

The BERT model significantly improved in all the evaluation metrics when fine tuned.

## Comparing word2vec and BERT

Word2vec generates one single vector for a word, whereas BERT generates more than one vector for a word, depending on the context of the word. We can utilize the contextual information using the BERT model.

## References

[1] “Amazon fine food reviews.” https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews (accessed Nov. 02, 2022).

[2] “Quick tour.” https://huggingface.co/docs/transformers/quicktour (accessed Nov. 02, 2022).