**Rating Predictions for Reviews in Online Markets**

Avinash Komatineni

School of Computing,

Clemson University

Clemson, SC 29631

[akomati@g.clemson.edu](mailto:akomati@g.clemson.edu)Sathwika Adimulam

School of Computing,

Clemson University

Clemson, SC 29631

[asathwi@g.clemson.edu](mailto:asathwi@g.clemson.edu)

**APPLIED DATA SCIENCE CPSC 6300: SPRING 2023**

Keerthana Komatineni

School of Computing,

Clemson University

Clemson, SC 29631  
 [kkomati@g.clemson.ed](mailto:kkomati@g.clemson.ed)u

**ABSTRACT**

In this study, we explore the prediction of multi-dimensional ratings in online markets, including customer reviews, seller ratings, product ratings, and shipment ratings. Our analysis is based on a dataset obtained from a popular online market, which includes a diverse range of features, such as product descriptions, customer reviews, seller information, shipment details, and other relevant data. The present report discusses the findings of our comprehensive data analysis, which involved data collection, preprocessing, exploratory data analysis (EDA), and model development. We performed detailed analysis and feature extraction on the textual data in customer reviews using advanced machine learning techniques. Additionally, we conducted feature engineering to identify relevant features, such as seller reputation, product characteristics, and shipment details, that have a significant impact on ratings. We developed and evaluated a robust prediction model using various machine learning algorithms, including regression, decision trees, and ensemble methods, to accurately predict multi-dimensional ratings.

**Keywords**

Sentiment Classification; Review text; Multilabel Classification; Word2Vec [1]; BERT; Text Classification; Classifier; Neural Networks;

# INTRODUCTION

The growing popularity of online markets has made ratings and reviews critical in influencing consumer decisions. However, many user reviews in online markets often provide only a single overall rating or product rating. This can be misleading, as users may have specific complaints related to different components such as the product, seller, or shipping. For instance, if a user receives a product late, it is a delivery issue and should only affect the delivery rating. Having only one rating for the review may lead to incorrect perceptions about the product for new users.

To address this ambiguity, a potential solution is to provide separate ratings for each component, such as product-specific rating, seller-specific rating, and delivery-specific rating. This approach would enable new users to better understand the ratings, and it would also assist sellers, manufacturers, and logistics departments in identifying areas for improvement. Along with customer reviews, users can rate sellers, products, and shipment experiences, resulting in multi-dimensional ratings. Accurate prediction of these ratings is crucial for online marketplaces, sellers, and consumers to make informed decisions and ensure a satisfactory buying experience. Our study focuses on developing a comprehensive prediction model for multi-dimensional ratings in online markets. We will utilize a dataset from a popular online marketplace, Amazon, with 1249 observations from the year 2015. This dataset encompasses a wide range of information, including product descriptions, customer reviews, seller information, shipment details, and other pertinent data. Our approach will involve meticulous data analysis, feature extraction, and model development to achieve accurate ratings prediction across multiple dimensions.

It will also analyze customer reviews and other relevant data to identify key features that impact ratings, followed by developing and evaluating a robust prediction model using various machine learning algorithms. The findings of this investigation have the potential to provide valuable insights for online marketplaces and sellers in improving their offerings and enhancing the customer experience.

# SUMMARY OF EDA

The dataset utilized in this study comprises online Amazon product reviews, consisting of 1249 rows and 12 columns for the year 2015. Initially, the dataset did not contain product rating, seller rating, and shipping rating, so we manually added these columns by analyzing the reviews within our dataset. The columns in the dataset encompass Marketplace, Customer Id, Review Id, Product Id, Product Title, Product Category, Star Rating, Helpful votes, total votes, Review headline, Review body, Review date, Product rating, Seller rating and Shipping rating. The data covers a span of one year.

We excluded certain columns that are not relevant for predicting ratings, such as Marketplace, Customer Id, Review Id, Product Id, Product Title, Product Category, Helpful votes, Total votes, and review date. The target variables for prediction encompass review rating, product rating, seller rating, and shipping rating. We can use either review body or both review body and review headline as predictors to forecast the target variables. In order to select an appropriate model for our response variables, we will take into consideration their type and distribution. Given that the response variables are categorical and range from 1 to 5, we will employ multi-label classification methods to predict multiple categories or labels for a given input, as these methods are well-suited for rating and review prediction tasks. Models such as decision trees, random forests [2], gradient boosting classifiers [3] and Convolutional Neural Networks (CNN) will be utilized, as they are suitable for predicting categorical response variables.

To guarantee the accuracy, reliability, and robustness of our data and models, we performed an outlier analysis on the prediction variables using the IQR (interquartile range) method. The purpose of this analysis was to detect and address any potential data errors. As seen below, upon completion of the outlier analysis, we observed that no outliers were identified in the data, indicating that there is no need to remove any observations from the dataset.

Chart

Description automatically generatedShape, rectangle

Description automatically generated

Shape, square

Description automatically generatedChart

Description automatically generated

**Figure 1. Box Plot showing the data distribution**

In order to better understand the relationships between the prediction variables, we generated a correlation matrix heatmap [4]. This visualization allowed us to assess the pairwise correlations between the variables. As shown in the heatmap below, we observed a strong positive correlation between the review rating, product rating, and seller rating. However, the shipment rating showed a weak correlation with the other variables. This information provides valuable insights into the interdependencies among the variables and can help guide our modeling efforts.

Graphical user interface

Description automatically generated

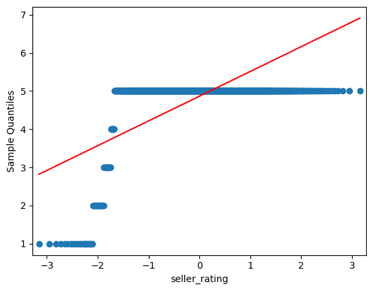
**Figure 2. Correlation Matrix Heatmap**

In order to assess the normality assumption of the prediction variables in our dataset, we utilized Q-Q plots. These plots offer a visual comparison between the data distribution and a normal distribution. The data may have a normal distribution if the data points fall in a straight line. Upon examining the Q-Q plots for the product rating and review rating variables, we observed that the data points aligned closely to a straight line, indicating that these variables may follow a normal distribution.

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

**Figure 3. Normality Assumption**

To assess the linearity and homoscedasticity assumptions, we made scatter plots to show the connection between the independent variable and the dependent variable. By examining scatter plots and inspecting the residuals for any discernible patterns, we aimed to determine if these assumptions hold true in our dataset. This analysis provides valuable insights into the nature of the relationship between the variables and helps to ensure the robustness of our models.

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

**Figure 4. Linearity and homoscedasticity assumptions**

# MACHINE LEARNING MODELS

In our analysis, we extensively evaluated various models and methods to predict the outcomes of our dataset. We evaluated these model’s performance using important measures like accuracy, precision, recall, and F1-score. Through careful evaluation, we identified the best-performing models. As a result, we have selected four models for our prediction purposes. The following models were employed in our analysis:

* Gradient Boosting Classifier
* Random Forest Classifier
* Decision Tree Classifier
* CNN

These models were chosen based on their performance and suitability for our specific dataset and prediction goals.

## Gradient Boosting Classifier

The Gradient Boosting Classifier (GBC) is a robust and powerful ensemble model that employs review text as input to predict multiple ratings simultaneously, including review rating, seller rating, product rating, and shipping rating. Through iterative combinations of multiple weak models, the Gradient Boosting Classifier creates a strong model capable of capturing complex relationships between the textual features and the different ratings. This makes it an ideal choice for prediction tasks that involve multiple ratings based on review text, offering high accuracy and predictive performance.

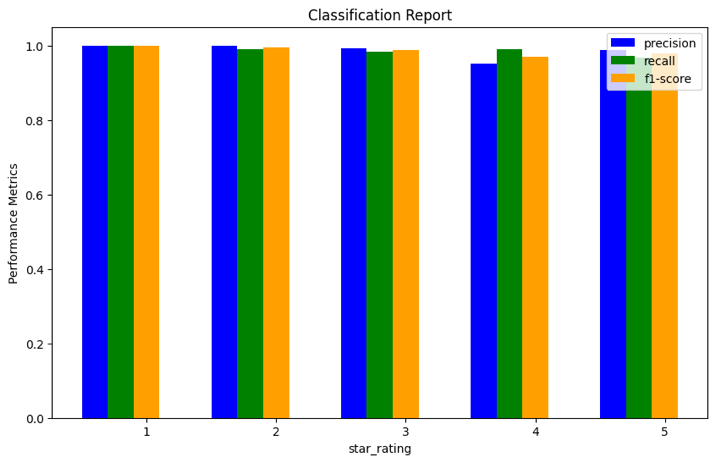
In our analysis, we utilized the Gradient Boosting Classifier model to predict the various ratings based on review text as input. To thoroughly evaluate its performance, we employed five different evaluation methods CountVectorizer, Tfidf Vectorizer [5], Word2Vec [6], Glove [7] and BERT, including accuracy, precision, recall, and F1-score, which collectively provided a comprehensive assessment of its predictive capabilities.

Moreover, the Gradient Boosting Classifier is well-regarded for its proficiency in handling imbalanced data. Its capacity to capture non-linear relationships and effectively handle large feature sets further enhances its effectiveness in simultaneously predicting multiple ratings. These compelling factors position it as a promising choice for our prediction purposes, and we have identified it as one of the top-performing models in our analysis. The table below showcases the model's accuracy, which was evaluated using five different methods.

**Table 1. Model Accuracy Evaluation for Gradient Boosting Classifier**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vectorizer | Precision | Recall | F1-Score | Accuracy |
| CountVectorizer | 0.78 | 0.764 | 0.77 | 0.65 |
| Tfidf Vectorizer | 0.74 | 0.79 | 0.77 | 0.775 |
| Word2Vec | 0.84 | 0.83 | 0.835 | 0.8 |
| Glove | 0.85 | 0.85 | 0.85 | 0.85 |
| BERT | 0.867 | 0.87 | 0.86 | 0.857 |

The image below shows how accurate the BERT model is when used with the gradient boosting classifier. The accuracy metric was used to evaluate how well the model can predict ratings based on review text input, and the results are visually presented in the image. This graphical representation gives insights into the performance of the BERT model and highlights its effectiveness in predicting ratings.



**Figure 5. Bar Graph to show performance metrics of review rating**

Chart, bar chart

Description automatically generated

**Figure 6. Bar Graph to show performance metrics of shipping rating**

**Chart, bar chart

Description automatically generated**

**Figure 7. Bar Graph to show performance metrics of seller rating**

**Chart, bar chart

Description automatically generated**

**Figure 8. Bar Graph to show performance metrics of product rating**

Upon comparing the accuracy of the BERT model with other models using various evaluation methods, it was observed that the BERT model exhibited superior performance with 0.86 accuracy. This suggests that the BERT model outperformed other models in accurately predicting the ratings based on review text input.

## Random Forest Classifier

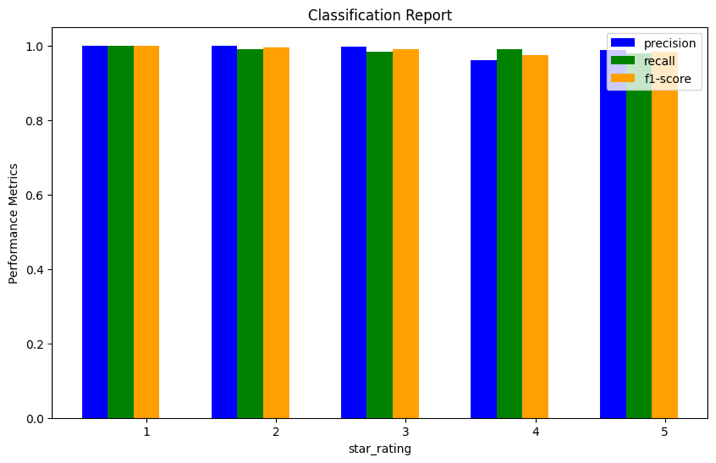
The Random Forest Classifier (RFC) is also an ensemble algorithm that uses review text as input to predict multiple scores simultaneously. It accomplishes this by building a collection of decision trees, each trained on a random selection of dataset samples and feature values. A final forecast is then made by combining the projections of various trees, creating a robust and reliable model. In our analysis, we utilized five distinct evaluation techniques - CountVectorizer, Tfidf Vectorizer [8], Word2Vec, Glove [9], and BERT - to assess the Random Forest Classifier's predictive capabilities. These methods involved measuring accuracy, precision, recall, and F1-score. Together, they provided a comprehensive assessment of the model's performance.

The Random Forest Classifier [10] has several advantages, including its ability to handle high-dimensional and complex data. Additionally, the ensemble nature of the algorithm helps to reduce overfitting and improve generalization performance. Hence, the Random Forest Classifier [11] emerges as a promising choice for our prediction purposes. These findings guarantee the suitability of the Random Forest Classifier for our analysis and underscore its potential as a reliable model for predicting multiple ratings based on review text input. The following table presents the accuracy of the model, which was assessed using five different techniques.

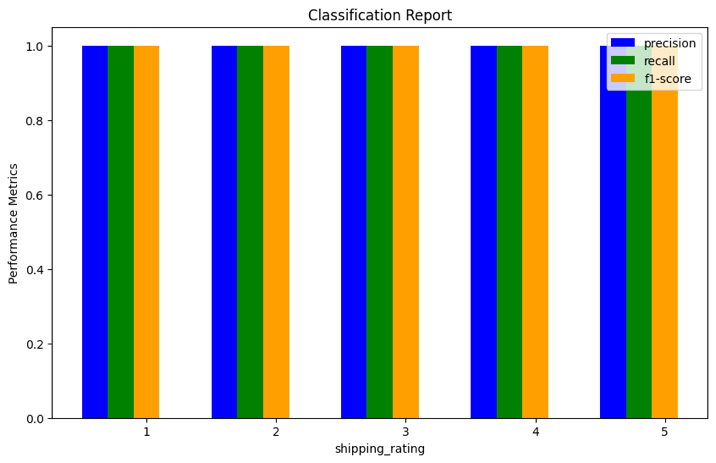
**Table 2. Model Accuracy Evaluation using Random Forest Classifier**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vectorizer | Precision | Recall | F1-Score | Accuracy |
| CountVectorizer | 0.785 | 0.765 | 0.77 | 0.65 |
| Tfidf Vectorizer | 0.79 | 0.81 | 0.799 | 0.78 |
| Word2Vec | 0.896 | 0.887 | 0.891 | 0.889 |
| Glove | 0.856 | 0.854 | 0.854 | 0.85 |
| BERT | 0.84 | 0.857 | 0.848 | 0.85 |

The visual depiction below illustrates the accuracy of the Word2Vec model in conjunction with the random forest classifier. The accuracy metric was employed to evaluate the model's ability to predict ratings based on review text input, and the results are visually presented in the image.



**Figure 9. Bar Graph to show performance metrics of review rating**

****

**Figure 10. Bar Graph to show performance metrics of shipping rating**

**Chart, bar chart

Description automatically generated**

**Figure 11. Bar Graph to show performance metrics of seller rating**

**Chart, bar chart

Description automatically generated**

**Figure 12. Bar Graph to show performance metrics of product rating**

When comparing the accuracies of the Random Forest Classifier using Word2Vec and BERT methods, we observed that both methods performed high and similarly, with a slight difference in accuracy. Interestingly, the Word2Vec method showed higher accuracy compared to the BERT method. Further analysis and experimentation could be conducted to investigate the reasons behind the superior performance of the Word2Vec method and to explore potential optimizations for other methods to achieve similar accuracy levels.

## Decision Tree Classifier

For classification tasks, the Decision Tree Classifier (DTC) is a popular machine learning technique. The data is divided into subsets using a recursive process depending on the input features, and decisions are based on the one of the input features that gives the better split at each node of the tree. The recursive process continues till it reaches the maximum depth or if there are minimum sample at a node. This method makes the Decision Tree Classifier a popular option for numerous classification tasks in machine learning because it successfully captures complex correlations between input data and the target variable.

The Decision Tree Classifier offers several advantages, making it a versatile choice for various applications. One notable advantage is its ability to handle both numerical and categorical data, making it suitable for diverse datasets. Additionally, the Decision Tree Classifier can capture complex patterns in the data, including nonlinear relationships between features. This makes it well-suited for tasks where the relationships between input features and output ratings are not linear, allowing for more accurate predictions in such scenarios.

In our study, the Decision Tree Classifier was employed to predict multiple ratings, including review rating, seller rating, product rating, and shipping rating, using review text as input. To assess the performance of the model, we employed five different methods for text representation: CountVectorizer, TfidfVectorizer, Word2Vec, Glove, and BERT. These methods offer distinct approaches for representing text data, each with their own strengths and weaknesses. By utilizing multiple methods, we were able to comprehensively evaluate the predictive capabilities of the Decision Tree Classifier model and provide a robust assessment of its performance. The results of our evaluation, summarizing the accuracy of the model using the different methods, are presented in the table below.

**Table 3. Model Accuracy Evaluation using Decision Tree Classifier**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vectorizer | Precision | Recall | F1-Score | Accuracy |
| CountVectorizer | 0.785 | 0.765 | 0.77 | 0.65 |
| Tfidf Vectorizer | 0.79 | 0.81 | 0.799 | 0.78 |
| Word2Vec | 0.867 | 0.86 | 0.86 | 0.864 |
| Glove | 0.856 | 0.854 | 0.854 | 0.85 |
| BERT | 0.84 | 0.857 | 0.845 | 0.85 |

The following image provides a graphical representation of the accuracy achieved by the Word2Vec model.

Chart, bar chart

Description automatically generated

**Figure 13. Bar Graph to show performance metrics of review rating**

**Chart, bar chart

Description automatically generated**

**Figure 14. Bar Graph to show performance metrics of shipping rating**

**Chart, bar chart

Description automatically generated**

**Figure 15. Bar Graph to show performance metrics of seller rating**

**Chart, bar chart

Description automatically generated**

**Figure 16. Bar Graph to show performance metrics of product rating**

When comparing the accuracy of the Decision Tree Classifier using Word2Vec and BERT methods, we found that both methods performed well with a slight difference in accuracy. Surprisingly, the Word2Vec method showed higher accuracy compared to the BERT method.

* 1. **Convolutional Neural Networks**

CNNs are a type of deep neural network commonly used in image processing tasks, but they have also been adapted for natural language processing tasks such as text classification. In text classification, CNNs [15] can be used to learn features from the text data by using convolutional filters, which slide over the input text and extract local features. These features are then passed through pooling layers to obtain a fixed-size representation of the text, which can be fed into a fully connected layer for classification.

One advantage of using CNNs for text classification is that they can learn hierarchical features from the text data. For example, lower-level features such as character and word-level patterns can be learned by the initial convolutional layers, while higher-level features such as phrases and sentence-level patterns can be learned by subsequent convolutional and pooling layers. This allows the model to capture both local and global patterns in the text data.

To apply CNNs [16] to classify text, we utilized five distinct approaches for text representation: CountVectorizer, TfidfVectorizer, Word2Vec, Glove, and BERT, each with their own strengths and weaknesses. By utilizing multiple methods, we comprehensively evaluated the predictive capabilities of the CNN model and provided a robust assessment of its performance. The table below summarizes the results of our evaluation, which presents the accuracy of the model using the different text representation methods.

**Table 4. Model Accuracy Evaluation using CNN**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vectorizer | Precision | Recall | F1-Score | Accuracy |
| CountVectorizer | 0.785 | 0.765 | 0.77 | 0.65 |
| Tfidf Vectorizer | 0.79 | 0.81 | 0.799 | 0.78 |
| Word2Vec | 0.87 | 0.872 | 0.870 | 0.869 |
| Glove | 0.86 | 0.81 | 0.834 | 0.82 |
| BERT | 0.84 | 0.863 | 0.851 | 0.86 |

In the evaluation of the CNN's accuracy, the Word2Vec method outperformed the other methods.

## 3.5 Test Error Statistics

To analyze the performance of the four models, we are using the following metrics [12] as standard statistics:

1. Precision: A classification model's precision is a measure of how well it correctly predicts positive cases out of all the instances that are anticipated to be positive. It is calculated as TP / (TP + FP), which is the ratio of true positives (TP) to the total number of true positives and false positives (FP) [13]. Precision reflects the model's ability to minimize false positives, or instances wrongly predicted as positive.

2. Recall: A classification model's recall, sometimes referred to as sensitivity or true positive rate (TPR), is a measurement of how well it correctly predicts positive cases out of all the actual positive instances. It is determined as follows: TP/ (TP + FN), where TP is the number of true positives and FN is the total of true positives plus false negatives. Recall reflects the model's ability to minimize false negatives, or instances wrongly predicted as negative.

3. F1-Score: The F1-score [14] is a metric that provides a balanced assessment of a model's performance by combining precision and recall into a single value. It is determined using the formula 2 \* (precision \* recall) / (precision + recall) as the harmonic mean of precision and recall. The F1-score is a helpful metric for assessing the overall effectiveness of a model because it is frequently employed when precision and recall are crucial and must be balanced.

4. Accuracy: A classification model's accuracy refers to how effectively it predicts every instance, both positive and negative, with accuracy. It is computed as the proportion of accurately predicted cases both true positives and true negatives to the total number of instances. Accuracy is commonly used as a metric for binary classification tasks but may not be suitable for imbalanced datasets where the proportions of classes are significantly different.

**Table 4. Most Accurate Methods**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **BEST Method** | **Precision** | **Recall** | **F1-Score** | **Accuracy** |
| **GBC** | **BERT** | **0.867** | **0.87** | **0.86** | **0.857** |
| **RFC** | **Word2Vec** | **0.896** | **0.887** | **0.891** | **0.889** |
| **DTC** | **Word2Vec** | **0.867** | **0.86** | **0.86** | **0.864** |
| **CNN** | **Word2Vec** | **0.87** | **0.872** | **0.870** | **0.869** |

# SUMMARY

The report presents a comprehensive evaluation of four different classification models, namely Decision Tree Classifier, Gradient Boosting Classifier, and Random Forest Classifier and Neural Networks for predicting multiple ratings using review text as input. Five different methods for representing text data, including CountVectorizer, TfidfVectorizer, Word2Vec, Glove, and BERT, were utilized to assess the accuracy and performance of these models.

The results showed that all four models performed well in predicting review rating, seller rating, product rating, and shipping rating. The Word2Vec method consistently demonstrated higher accuracy compared to the other methods for all four models. However, there were slight variations in performance across the models, with Gradient Boosting Classifier and Random Forest Classifier showing slightly higher accuracy compared to Decision Tree Classifier and Neural Networks in some cases.

The comparison and evaluation of various text data representation techniques, especially Word2Vec and BERT, in the context of four different classification models Decision Tree Classifier, Gradient Boosting Classifier, and Random Forest Classifier and Neural Networks is the main contribution of this study. By conducting a systematic analysis and experimentation, this study has provided insights into the performance of these methods in terms of accuracy for the given dataset. The findings of this research contribute to the existing literature on text data representation and classification by showcasing the relative performance of different methods, and shedding light on the potential benefits of using Word2Vec as compared to BERT in the context of the studied models. This research also highlights the importance of considering different methods of text data representation when developing classification models for text-based tasks and provides a foundation for further exploration and optimization of these methods in future research.

Together, the Random Forest Classifier with Word2Vec embeddings is a powerful combination for rating prediction problems, as it leverages the strengths of both techniques. The Random Forest provides robustness and flexibility in handling complex data, while the Word2Vec embeddings enhance the model's ability to capture semantic meaning from text data. This model can be used in various applications such as sentiment analysis, recommendation systems, and feedback analysis, where predicting ratings accurately is crucial for making informed decisions. Although all other models and vectorizers work almost like the above combination, Random Forest Classifier stood above a bit in terms of performance.

Overall, the report provides a detailed evaluation of the performance of models in predicting ratings using review text data, considering various text representation methods. The results highlight the importance of selecting an appropriate text representation method and model for the specific task at hand and provide valuable insights for further analysis and optimization of the models.

In conclusion, this research investigated and compared the accuracies of Decision Tree Classifier, Gradient Boosting Classifier, Random Forest Classifier and Neural Networks using Word2Vec and BERT methods for text data representation. The findings of this study revealed that all performed well with a slight difference in accuracy, with Word2Vec for random forest classifier showing higher accuracy compared to BERT. This suggests that Word2Vec may be a more effective method for text data representation in the context of the studied classification models. However, further analysis and experimentation could be conducted to investigate the reasons behind the superior performance of Word2Vec, and to explore potential optimizations for other methods to achieve similar accuracy levels. Overall, this research contributes to the understanding of text data representation in the context of classification models and provides insights that can inform future research in this area.

Github Repository Link :

# REFERENCES

1. <https://medium.com/data-science-lab-spring-2021/amazon-review-rating-prediction-with-nlp-28a4acdd4352>
2. <https://escholarship.org/uc/item/0613c9z7>
3. <https://peerj.com/articles/cs-914/>
4. https://medium.com/@arhamrumi/amazon-reviews-eda-662c485ec00c
5. <https://link.springer.com/article/10.1007/s11761-019-00278-6>
6. <https://wiki.pathmind.com/word2vec>
7. <https://medium.com/the-innovation/sentiment-analysis-using-word2vec-and-glove-embeddings-5ad7d50ddb0d>
8. <https://www.analyticsvidhya.com/blog/2020/02/quick-introduction-bag-of-words-bow-tf-idf/>
9. <https://blogs.sap.com/2019/07/03/glove-and-fasttext-two-popular-word-vector-models-in-nlp/>
10. <https://www.kaggle.com/code/jonathanoheix/sentiment-analysis-with-hotel-reviews/notebook>
11. <https://link.springer.com/article/10.1007/s10115-012-0600-z>
12. <https://medium.com/analytics-vidhya/performance-metrics-precision-recall-f1-score-efb51ac111bd>
13. <https://towardsdatascience.com/a-look-at-precision-recall-and-f1-score-36b5fd0dd3ec>
14. <https://vitalflux.com/accuracy-precision-recall-f1-score-python-example/>
15. <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9226403>
16. https://www.sciencedirect.com/science/article/abs/pii/S0167865518303325