Individual Coursework Submission Form

Specialist Masters Programme

Please list all names of group members: 1. Rawal, Nikhil (230044335) 2. Verma, Shubham (230021430) 3. Sawant, Pranav (230062637) 4. Raju Mysore Yathindra Raju, Suhruth (230041328)	**GROUP NUMBER:** **14**
MSc in: Business Analytics	
Module Code: SMM694	
Module Title: Applied Natural Language Processing (NLP)	
Lecturer: Dr. Yijun Wang	**Submission Date:** 20 July, 2024

Declaration:

By submitting this work, we declare that this work is entirely our own except those parts duly identified and referenced in my submission. It complies with any specified word limits and the requirements and regulations detailed in the coursework instructions and any other relevant programme and module documentation. In submitting this work we acknowledge that we have read and understood the regulations and code regarding academic misconduct, including that relating to plagiarism, as specified in the Programme Handbook. We also acknowledge that this work will be subject to a variety of checks for academic misconduct.

We acknowledge that work submitted late without a granted extension will be subject to penalties, as outlined in the Programme Handbook. Penalties will be applied for a maximum of five days lateness, after which a mark of zero will be awarded.

Marker's Comments (if not being marked on-line):

Deduction for Late Submission: | **Final Mark:** **%**

1. INTRODUCTION

The objective of this project is to develop a predictive model that accurately categorises beer reviews from the BeerAdvocate website into one of three distinct rating categories: “okay”, “good”, and “excellent”. These ratings are defined as follows:

- 0/okay: Rating is in the range [3.5 - 4)
- 1/good: Rating is in the range [4 - 4.5)
- 2/excellent: Rating is in the range [4.5 - 5]

User reviews are crucial for consumers in making informed purchasing decisions, as they provide insights and opinions from previous buyers. In the realm of e-commerce and consumer products, the ability to automatically classify and summarise user reviews can significantly enhance user experience and assist in product selection.

To achieve this goal, we employed various machine learning models to determine which classifier performs best on our dataset. By leveraging natural language processing (NLP) techniques and different feature extraction methods, we aimed to build robust models capable of accurately predicting the ratings based on review text alone. The models evaluated include Logistic Regression, Support Vector Machines (SVM), Random Forests, and Naive Bayes classifiers, each trained and validated using TF-IDF vectorisation techniques.

This project underscores the importance of machine learning in text classification tasks, highlighting how advanced algorithms can transform unstructured textual data into meaningful and actionable insights. Through this analysis, we strive to identify the most effective model that can be deployed for real-world applications, aiding both consumers and businesses in navigating the vast array of product reviews available online.

2. METHODOLOGY

2.1 Dataset

The training dataset contains 21,057 labeled reviews. The first column, “label”, is the label of each training beer review; the second column, “text”, contains the text of the beer review. The test dataset contains 8,943 unlabeled reviews. The first column, “id”, is a beer review-level identifier; the second column, “text”, is the unlabelled beer review to classify. The test label dataset contains the first column, “id”, which is the review-level identifier reported in test data, and the second column, “pred label”, contains the predicted class label (0, 1, or 2) for the corresponding line in the text data.

2.2 Exploratory Data Analysis

Class Distribution: We examined the distribution of ratings to determine if there was any class imbalance. As illustrated in Figure 1, the distribution of rating classes is balanced. This step is crucial as a balanced dataset ensures that the model does not become biased towards any class.

Word Clouds: Word clouds were generated for each rating category to visualise the most frequently occurring words. These word clouds revealed common terms and themes that differentiate the categories. Appendix A shows the word cloud for each of the rating categories. Specific adjectives and descriptors that are more prevalent in positive or negative reviews were identified, providing insights into the characteristics of each rating category.

Review Length Distribution: We analysed the distribution of review lengths across the different rating categories. This analysis, depicted in Figure 2, shows how the length of the review text varies among the categories. The results indicate whether longer or shorter reviews are more common in certain categories, which could reflect the level of detail or expressiveness associated with different ratings.

Top Bigrams: The identification of the most common bigrams (two-word phrases) in the reviews provided further insights. As shown in Figure 3, these key phrases often carry important contextual information that single words might miss. This analysis helps in understanding the frequent combinations of words that occur in reviews and can inform feature engineering for the classification model.

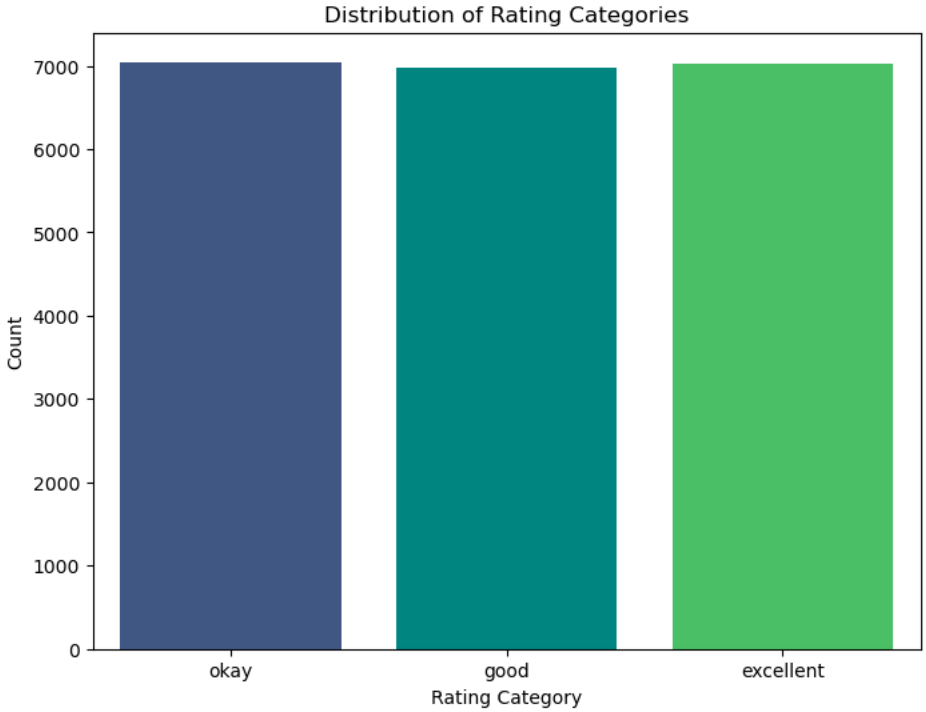

Figure 1. Class Distribution in the Dataset

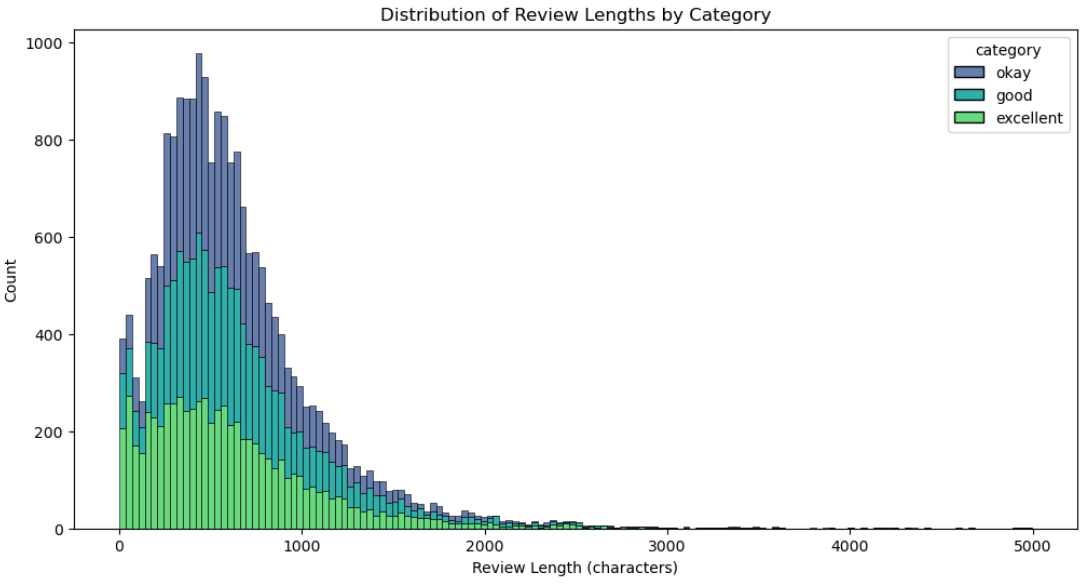

Figure 2. Distribution of Review lengths by rating category

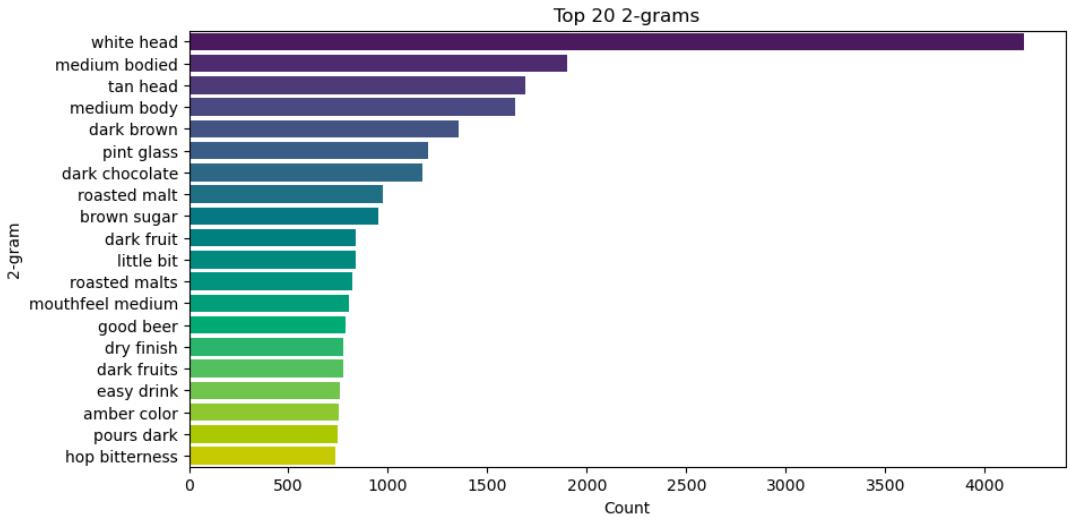

Figure 3. Top 20 - bi Grams

2.3 Data Preprocessing

Data preprocessing is an essential step in preparing a dataset for machine learning applications. Our preprocessing pipeline involved several stages, including text cleaning and feature extraction.

Text Cleaning: The text cleaning process involved removing stop words and punctuation marks. Stop words, such as "the," "and" and "is," are commonly used words that do not contribute significant meaning to the text analysis and can clutter the data. Punctuation marks were removed to ensure consistency in text tokenisation and to avoid noise in the data, which can hinder model performance.

We also performed lemmatisation and tokenisation. Lemmatisation reduces words to their base or root form, which helps in normalising the text and reducing inflectional forms, ensuring that words like "running" and "runs" are treated as the same token "run." Tokenisation, on the other hand, splits the text into individual words or tokens, making it easier to analyse and process the text data.

Feature Extraction:

Feature extraction transforms raw text into numerical features suitable for machine learning models. We utilized the TF-IDF (Term Frequency-Inverse Document Frequency) technique to convert the text data into numerical features. TF-IDF is a widely used method that reflects the importance of a word in a text document relative to its occurrence across the entire document corpus.

TF-IDF was chosen over other methods like Word2Vec and Bag of Words (BoW) due to its balance of informativeness and computational efficiency. While BoW treats all words equally, potentially diluting the significance of key terms, and Word2Vec captures semantic relationships but requires extensive computational resources and training time, TF-IDF effectively highlights important words by considering their frequency within documents and their rarity across the corpus. This approach provides a meaningful feature representation without the high computational cost associated with Word2Vec (Ramos, J. 2003), (Goldberg, Y., 2014).

2.4 ML Classifiers

2.4.1 Logistic Regression (LR)

Logistic Regression is a commonly used algorithm for binary and multi-class classification tasks, including text classification. It is preferred for its simplicity, interpretability, and efficiency in handling high-dimensional data, such as text data transformed through vectorisation techniques. According to Pedregosa et al. (2011), Logistic Regression, when combined with appropriate regularisation, can prevent overfitting and provide robust performance in text classification tasks.

2.4.2 Multinomial Naïve Bayes

Naive Bayes classifiers are particularly effective for text classification due to their strong assumptions of feature independence, which often hold true in practice despite the simplification. The algorithm is computationally efficient and works well with high-dimensional data, making it suitable for large text datasets. Rennie

et al. (2003) discuss how Naive Bayes can be adapted and optimised for text classification, highlighting its relevance and effectiveness in such tasks.

2.4.3 Support Vector Machine (SVM)

SVMs are powerful classifiers that work well with both linear and non-linear data using kernel functions. They are known for their robustness and effectiveness in high-dimensional spaces, such as text data. Joachims (1998) illustrates the application of SVMs in text categorization, demonstrating their ability to handle the complex feature spaces typically found in text classification problems.

3. RESULTS

3.1 Performance Evaluation Metrices

In our analysis, we assessed the performance of different classifiers using key metrics such as **Accuracy**, **Precision**, **Recall**, and the **F1 Score**. We prioritised F1 Score as it is a harmonic mean of precision and recall. Table 1 summarises the evaluation metrics for various classifiers, aiding in the selection of the Support Vector Machine model for this analysis.

3.2 Summary of Results

In our classifier comparison involving Logistic Regression, Multinomial Naïve Bayes, Support Vector Machine, we found that Support Vector Machine demonstrated superior performance across all metrics, as summarised in Table 1 of our results. This solidifies our choice of SVM as the preferred final model in our analysis. Figure 4 shows a visual representation of these results in the form of bar chats.

Table 1. Performance Comparison of ML Classifiers based on evaluation metrices.

Models	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.5990	0.5919	0.6001	0.5946
Naïve Bayes	0.5836	0.5760	0.5848	0.5782
Support Vector Machine	0.6076	0.6029	0.6085	0.6050

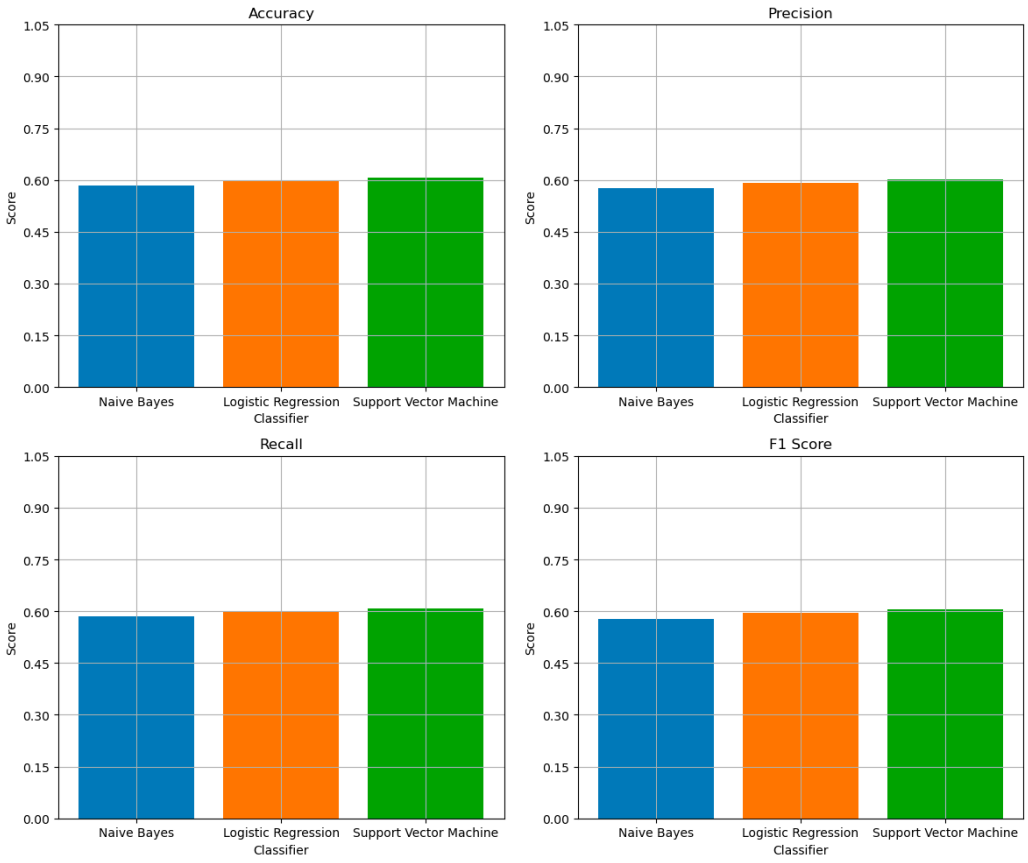

Figure 4. Test Results showing Accuracy, Precision, Recall and F1 Score

4. DISCUSSION

The results indicate that the Support Vector Machine (SVM) outperforms Logistic Regression, Naive Bayes, and Random Forest in terms of accuracy, precision, recall, and F1 score. This is likely due to the SVM's ability to handle high-dimensional data and effectively separate classes using hyperplanes.

While Logistic Regression and Naïve Bayes are efficient, interpretable, and known for their simplicity and speed, they may not capture complex relationships between features effectively. Random Forest, although robust and less prone to overfitting, did not perform as well as SVM in our specific text classification task, possibly due to the high-dimensional nature of the text data and the need for more sophisticated feature separation.

Implication of Findings:

The superior performance of SVM suggests that it is well-suited for complex text classification tasks. This model can be effectively deployed in real-world applications, enhancing automated review classification systems. Future work could explore advanced NLP techniques such as transformer-based models (e.g., BERT, GPT) to capture deeper semantic relationships within the text. Additionally, incorporating sentiment analysis and other contextual features could further improve model performance.

Limitations and Future Work

Despite the success of the SVM model, there are limitations to our current approach. Future work could involve experimenting with advanced embeddings such as BERT to potentially enhance model performance. Transfer learning using pre-trained models like BERT, GPT, or other transformer-based architectures could also be explored to leverage the vast amount of linguistic knowledge these models possess.

Another area for future exploration is the integration of sentiment analysis into the classification process. Sentiment analysis could provide additional contextual information that may enhance the model's ability to differentiate between the subtle nuances of "good" and "excellent" reviews.

Moreover, the current approach focuses solely on the textual content of the reviews. Including metadata such as user ratings, the date of the review, and other user-specific information could provide a more comprehensive understanding and improve classification accuracy. Studies have shown that integrating multiple sources of information can significantly enhance the performance of machine learning models in text classification tasks (Yang et al., 2016).

Lastly, improving the preprocessing pipeline to include advanced text normalization techniques such as named entity recognition (NER) and part-of-speech (POS) tagging

could further refine the feature extraction process. These techniques can help in identifying and preserving the semantic structure of the text, which is often lost during basic tokenization and lemmatization (Manning et al., 2008).

5. REFERENCES

- Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
- Pedregosa F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12, 2825-2830.
- Rennie J. D. M., Shih L., Teevan J., & Karger D. R. (2003). Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In Proceedings of the 20th International Conference on Machine Learning (ICML-03) (pp. 616-623).
- Joachims T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In European Conference on Machine Learning (pp. 137-142). Springer Berlin Heidelberg.
- Manning C. D., Raghavan P., & Schütze H. (2008). Introduction to Information Retrieval. Cambridge University Press.
- Mikolov T. et al. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.
- Goldberg Y., & Levy O. (2014). Word2Vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method. arXiv preprint arXiv:1402.3722.
- Ramos J. (2003). Using TF-IDF to Determine Word Relevance in Document Queries. In Proceedings of the First Instructional Conference on Machine Learning.
- Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In Proceedings of the Fourteenth International Conference on Machine Learning (ICML '97) (pp. 412-420). Morgan Kaufmann Publishers Inc.

APPENDIX A

Word Cloud for Okay Reviews

Word Cloud for Good Reviews

Word Cloud for Excellent Reviews