Yelp Dataset Challenge

DS3010 Final Project

Ethan Vaz Falcão, Emre Sabaz, Maanav Iyengar, Nur Fateemah, Sarah Kogan

Introduction

Online review platforms like Yelp have become increasingly popular as a primary source of information for users seeking recommendations on businesses, restaurants, and services. The Yelp dataset is a comprehensive collection of reviews and tips across various cities and categories. It consists of several files, including businesses, users, check-ins, reviews, and tips, all of which contain rich textual content and associated metadata, making it an ideal resource for analysis and modeling.

The business file provides information about the establishments listed on Yelp, such as their ID, name, address, categories, and geographic coordinates. Similarly, the user file provides details about registered Yelp users, including their ID, name, and the number of reviews they have written. The check-in file records the number of times a business has been checked into on Yelp, along with its ID. The review file contains the textual content of user reviews, along with their respective ratings and timestamps, as well as the IDs of the users and businesses. The tip file contains short tips or recommendations provided by users for various businesses, along with their respective IDs.

The Yelp dataset offers a comprehensive view of user-generated content and interactions within the platform. The Yelp Dataset Challenge provides data scientists with the opportunity to extract valuable insights and solve various challenges in the field of data science. In particular, this project aims to leverage the Yelp dataset to address two key problems: predicting business attributes using review and tip textual information, and detecting fake reviews using sentiment analysis and machine learning models.

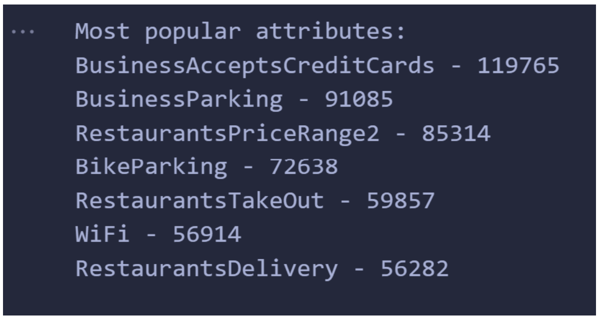
Task 1

Objective

The objective of this task is to predict the business attributes using review and tip textual information.

Data Preprocessing

In this study, the most prevalent binary attributes were selected, as these were deemed to be more pertinent to customers and compatible with logistic regression analysis. Based on the data presented in Figure 1, the attributes that emerged as the most common were BusinessAcceptsCreditCards, BikeParking, and RestaurantsTakeOut. To effectively analyze this data, vectorization techniques were employed to transform the textual information into numerical representations. Subsequently, logistic regression was utilized for the purpose of classification, enabling a more sophisticated and professional examination of these key attributes.



```
Most popular attributes:
BusinessAcceptsCreditCards - 119765
BusinessParking - 91085
RestaurantsPriceRange2 - 85314
BikeParking - 72638
RestaurantsTakeOut - 59857
WiFi - 56914
RestaurantsDelivery - 56282
```

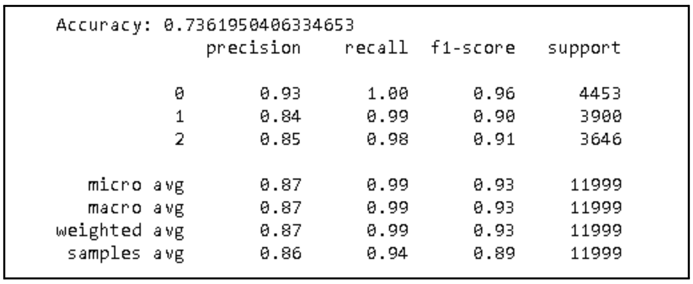
Figure 1 - Most Common Attributes

Training & Results

The overall accuracy rate achieved for the whole dataset was 0.74. The multi-label classification was applied to the data with review text and tip text as the features and three different business attributes as labels. The models we used were SVC, logistic regression, and Random Forest.

As shown in Figure 2, the precision scores for the classes are relatively high, with 0.93 for class 0, 0.84 for class 1, and 0.85 for class 2. These values indicate that the model had a low rate of false positives, correctly identifying instances belonging to these classes. The recall scores are also impressive, with values of 1.00 for class 0, 0.99 for class 1, and 0.98 for class 2. These scores show that the model had a low rate of false negatives, effectively capturing the instances from each class. The F1 scores, which consider both precision and recall, are reasonably good, with values of 0.96 for class 0, 0.90 for class 1, and 0.91 for class 2. Overall, the model demonstrates a strong performance, particularly for class 0, with many instances accurately classified. The micro-average accuracy, recall, and

F1-score is 0.87, 0.99, and 0.93, respectively, indicating good performance on a per-instance basis.



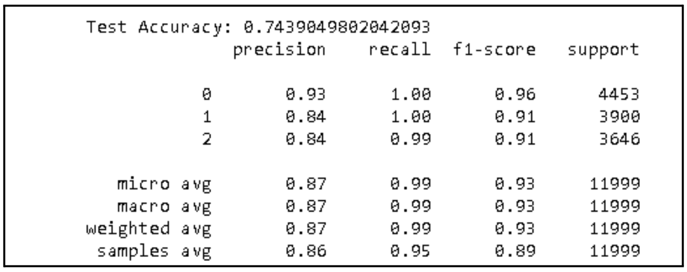
```
Accuracy: 0.7361950406334653
              precision    recall  f1-score   support

           0       0.93      1.00      0.96      4453
           1       0.84      0.99      0.90      3900
           2       0.85      0.98      0.91      3646

   micro avg       0.87      0.99      0.93     11999
   macro avg       0.87      0.99      0.93     11999
weighted avg       0.87      0.99      0.93     11999
 samples avg       0.86      0.94      0.89     11999
```

Figure 2 - Random Forest Model Classification Report For Task 1

As shown in Figure 3, The precision scores indicate the proportion of correctly predicted instances out of the total instances predicted for each class. For class 0, the precision is 0.93, indicating that 93% of the instances predicted as class 0 were correct. Similarly, for class 1, the precision is 0.84, and for class 2, it is also 0.84. The recall scores measure the proportion of correctly predicted instances out of the total instances of each class. Class 0 has a perfect recall score of 1.00, meaning that all instances belonging to class 0 were correctly identified. Class 1 and class 2 also have high recall scores of 1.00 and 0.99, respectively. The F1 scores, which consider both precision and recall, are 0.96 for class 0, 0.91 for class 1, and 0.91 for class 2, indicating a balanced performance.



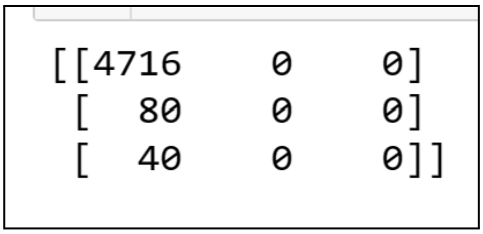
```
Test Accuracy: 0.7439049802042093
              precision    recall  f1-score   support

           0       0.93      1.00      0.96      4453
           1       0.84      1.00      0.91      3900
           2       0.84      0.99      0.91      3646

   micro avg       0.87      0.99      0.93     11999
   macro avg       0.87      0.99      0.93     11999
weighted avg       0.87      0.99      0.93     11999
 samples avg       0.86      0.95      0.89     11999
```

Figure 3 - SVC Classification Report for Task 1

The confusion matrix for the results of Task 1 are shown in Figure 4. -4616 true negative predictions for the 'BusinessAcceptsCreditCards' class, meaning that 4617 instances in the test set were correctly classified as not accepting credit cards for businesses. -80 false negative predictions for the 'BusinessAcceptsCreditCards' class, meaning that 94 instances in the test set were incorrectly classified as not accepting credit cards for businesses. -40 false negative predictions for the 'BusinessAcceptsCreditCards' class, meaning that 36 instances in the test set were incorrectly classified as not accepting credit cards for businesses.



```
[[4716    0    0]
 [  80    0    0]
 [  40    0    0]]
```

Figure 4 - Confusion Matrix for Task 1

The workflow for this task is shown in Figure 5.

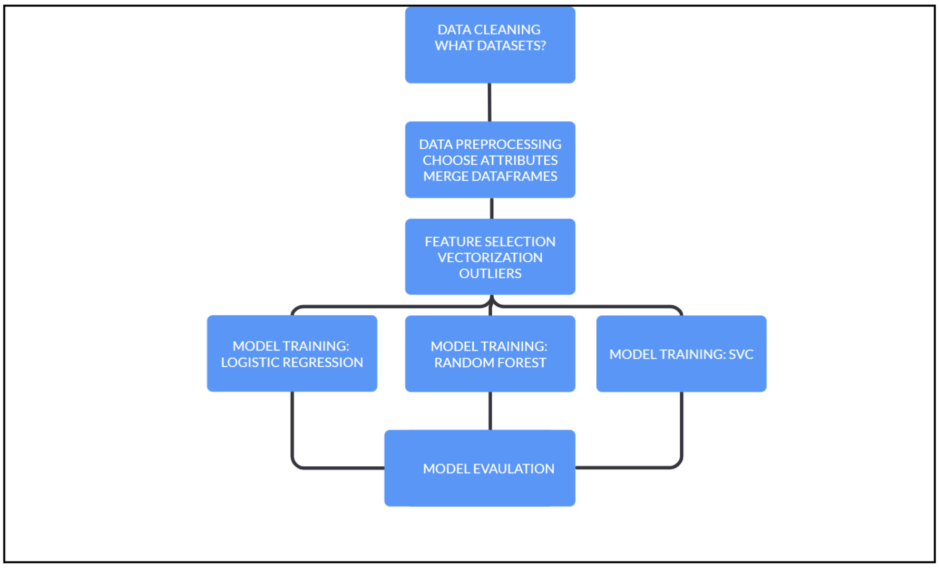


Figure 5 - Workflow for Task 1

Business Applications

There are two primary business applications that can be found when predicting the business attributes based on review and tip textual information: Targeted marketing and competitor analysis. For the former, businesses can use the insights from the model to tailor their marketing efforts towards specific target audiences. They can develop campaigns highlighting the attributes preferred by their customers, ultimately leading to higher customer satisfaction and increased revenue. For the latter, the model can help businesses gain insights into their competitors' performance and attributes, enabling them to adapt their strategies and offerings to stay ahead in the market.

Task 2

Objective

Our objective is to use a machine learning model to help accurately detect fake reviews in the Yelp dataset, to improve the reliability of online reviews.

Data Preprocessing

The initial phase of this project involved preprocessing and cleaning the data. A data frame of the review dataset, including the 'user_id', 'prod_id', and 'date' columns, was created. We then randomly sampled 1 million rows from the dataset to reduce the computational time required for subsequent analysis. The dataset was examined for missing values and none were found, ensuring data quality and integrity. Subsequently, general statistics of the data, such as the number of product IDs (044) and the number of unique users (260, 239), were calculated. Additionally, a label column was added to determine the sentiment analysis of each review. the determination of whether a review is fake or not is based on the sentiment analysis of the review text. Specifically, the sentiment score for each review is calculated using the VADER (Valence Aware Dictionary and sEntiment Reasoner) tool from the NLTK library. If the sentiment score is greater than or equal to 0.5, the review is considered positive and therefore not fake. Conversely, if the sentiment score is less than 0.5, the review is considered negative and potentially fake. It is important to note that this approach is based solely on the sentiment of the review and does not take into account other factors that could indicate a fake review, such as the language used or the reviewer's history. Additionally, the threshold of 0.5 for the sentiment score is arbitrary and may not be optimal for all scenarios. Therefore, while this approach can be useful, it should be used in conjunction with other methods to more accurately detect fake reviews.

In the feature engineering phase, crucial features for detecting fake reviews were identified, such as the maximum number of reviews in a day, the percentage of reviews with positive and negative ratings, review length, and the statistics of ratings for the reviewers' reviews. The average word length, number of sentences, average sentence length, and the percentage of numerals and capitalized words were also calculated. Afterward, the text was tokenized by breaking the strings into lists of words.

We used the data first through supervised learning, the models we used were a random forest classifier, gradient boosting classifier, and Logistic regression. We evaluated their F scores, accuracy rates (which were all above 80 percent), recall scores, and AUC score. We generated a confusion matrix for the highest model output which was gradient boosting. We also generated feature importance since we had many features that we compared to predict faker reviews. Since the model was unbalanced we had to use SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling). These are common ways to solve the issue of unbalanced data which is seen in previous Yelp dataset research papers.

Training & Results

Figure 6 shows the different models that use simple upsampling behavioral data, the Gradient Boosting model proved that it performed the best with the highest AUC score, accuracy score, recall score, and f1 score. These high scores are important because they indicate the model's ability to accurately classify both the majority and minority classes in the imbalanced dataset. The AUC (Area Under the Curve) score measures the overall performance of the model, considering the trade-off between true positive rate and false positive rate. A high AUC score suggests that the model has good discriminatory power.

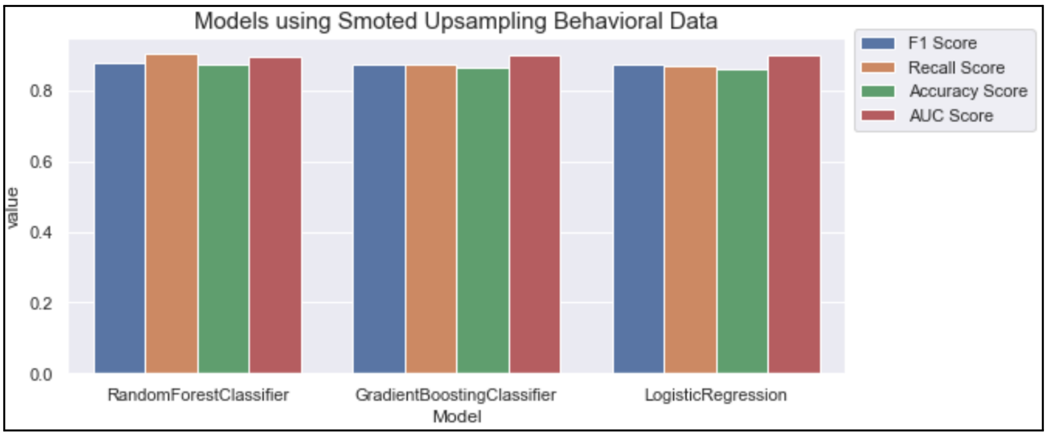


Figure 6 - Graph Showing Different Models using Smoted Upsampling Behavioral Data

Figure 7 shows the ROC curve for the fake review classification, both Smote and Adaysn have the same ROC area which is 0.90. The ROC curve is created by plotting the true positive rate (sensitivity) against the false positive rate (1 - specificity) at various classification thresholds. The area under the ROC curve (AUC) provides a measure of the model's ability to distinguish between the classes, with a higher AUC indicating better performance. In this case, an ROC area of 0.90 suggests that both SMOTE and ADASYN have successfully generated synthetic samples that help the models classify the minority class accurately. It indicates a strong discriminatory power, meaning that the models are effective in differentiating between positive and negative instances.

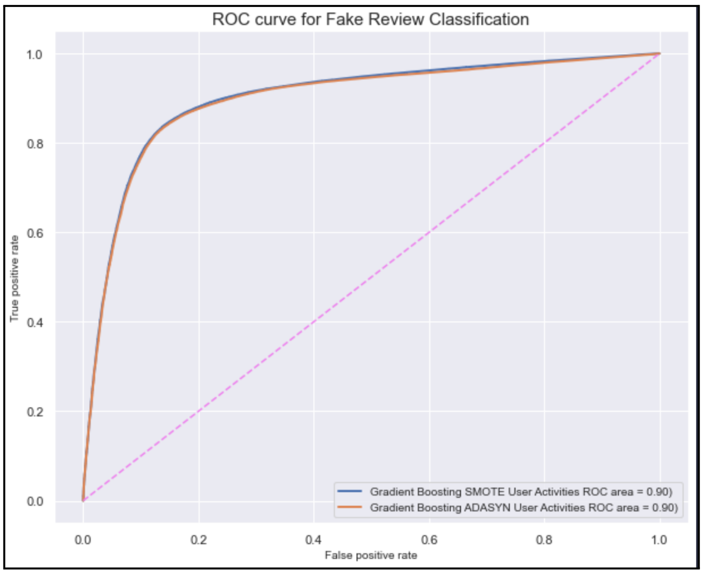


Figure 7 - ROC curve for Fake
Review Classification

Figure 8 shows the feature importance, and the model focused more on the stars rather than all the features equally, which is a downside of the model as it only placed importance on the stars.

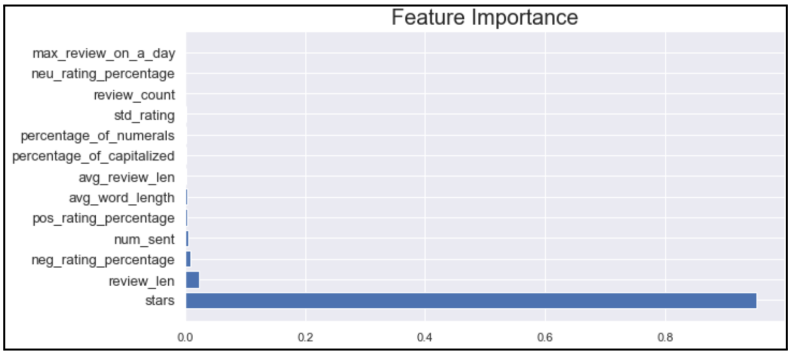


Figure 8 - Feature Importance

Figure 9 shows the confusion matrix of gradient boosting, it shows that the model has a higher number of false negatives (FN) compared to false positives (FP). This suggests that the model is more likely to incorrectly classify positive instances as negative. Overall, the evaluation metrics support the observations from the confusion matrix. The model demonstrates good accuracy, precision, and recall, indicating its ability to correctly classify instances. However, it is important to consider the relatively higher number of false negatives (21,209) and false positives (862) in the confusion matrix, which may impact the model's performance in certain scenarios.

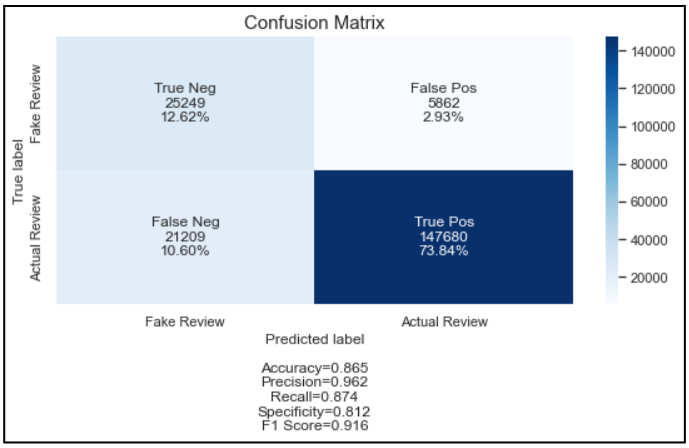


Figure 9 - Confusion Matrix of Gradient Boosting

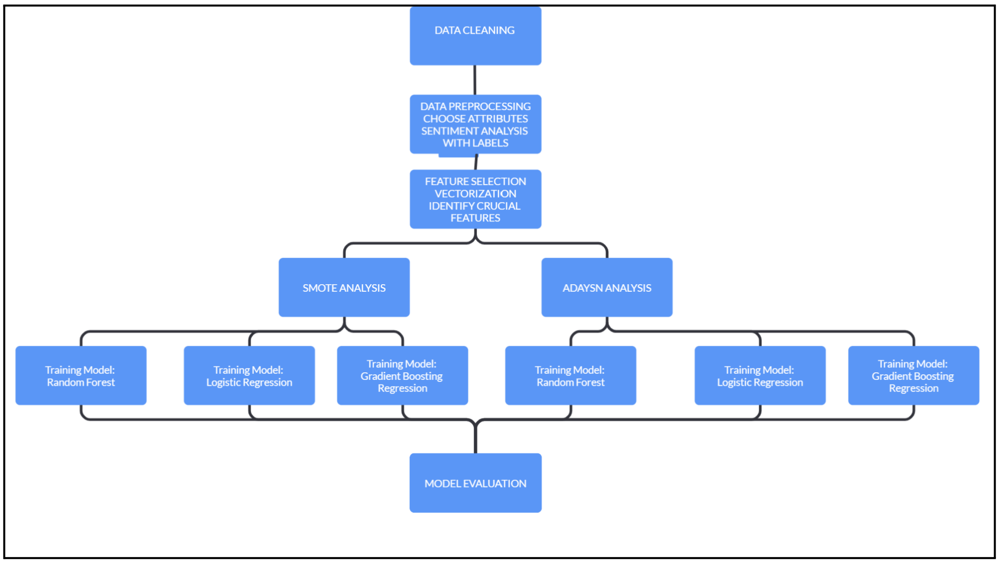

DATA CLEANING
DATA PREPROCESSING
CHOOSE ATTRIBUTES
SENTIMENT ANALYSIS
WITH LABELS
FEATURE SELECTION
VECTORIZATION
IDENTIFY CRUCIAL
FEATURES
SMOTE ANALYSIS
ADAYSN ANALYSIS
Training Model: Random Forest
Training Model: Logistic Regression
Training Model: Gradient Boosting Regression
Training Model: Random Forest
Training Model: Logistic Regression
Training Model: Gradient Boosting Regression
MODEL EVALUATION

Figure 10 - Task 2 Workflow

When using ADASYN (a technique for handling imbalanced datasets), the model's performance scores are spread out or distributed across different evaluation metrics. The random forest model has a higher recall and F-1 Score than gradient and logistic regression meaning, the random forest model shows better performance in correctly identifying positive instances and achieving a balance between precision and recall, but a lower Accuracy and AUC score meaning that, the random forest model might sacrifice overall accuracy and discriminative power for better performance in correctly identifying positive instances. Gradient boosting models and logistic regression have similar scores. The gradient boosting and logistic regression models exhibit similar performance scores across the evaluated metrics. It suggests that both models perform similarly in terms of accuracy, AUC score, recall, and F-1 Score. These models may have comparable abilities to classify instances and discriminate between the classes in the given context.

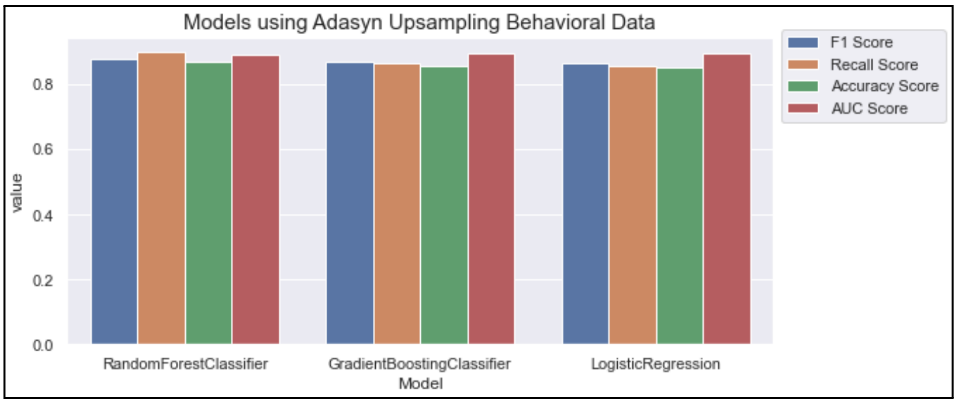


Figure 9 - Graph Showing Different Models using Adasyn Upsampling Behavioral Data

Business Applications

Four primary business applications can be found when predicting the business attributes based on review and tip textual information: Sentiment analysis, fraud detection, reputation management, and content moderation. First, by analyzing reviews and identifying fake ones, the model can help businesses understand genuine customer sentiment toward their products or services. This can aid in making data-driven decisions for product development, pricing, and customer service improvements. Second, the ability to identify fake reviews can help businesses and review platforms maintain their credibility and trustworthiness. By filtering out fake reviews, they can ensure that customers receive accurate information, leading to better-informed decisions. Third, by detecting and removing fake reviews, businesses can better manage their online reputation, preventing potential damage caused by misleading or malicious content. Finally, review platforms can utilize the model to automatically identify and filter out fake or low-quality reviews, ensuring that their users have access to reliable and high-quality information.

Conclusion

Challenges

For Task 1 and Task 2, there was a notable challenge due to the large volume of data, leading to memory constraints and runtime issues during testing with all datasets. To address this, the team opted to sample the data by selecting only a single prevalent year or 10,000 samples. Once all methods and code were finalized, the tests were run on a device with a higher RAM capacity to overcome the resource limitations.

Task 1 posed another obstacle, particularly in the division of attributes into separate columns and assigning them binary values. This necessitated meticulous consideration and attention to detail on the part of the team.

Similarly, Task 2 presented its own set of difficulties, particularly in determining the appropriate approach and models for accurately detecting fake reviews, given the inherent complexity of the task. To overcome this, extensive research was conducted on existing methods for predicting fake reviews.

To achieve the goal of developing a highly accurate machine capable of detecting all reviews, the team recognized the importance of addressing the class imbalance in the data before feeding it into the classifier. Upon analysis of the dataset, it was discovered that approximately 86% of the data represented true reviews, while 13% comprised false reviews.

To solve the class imbalance issue present in the dataset, the team employed the SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling) techniques. These methods are commonly used to address imbalanced data in machine learning tasks.

SMOTE works by generating synthetic samples of the minority class by interpolating between existing minority class samples. This technique helps in increasing the representation of the minority class and balancing the distribution of classes in the dataset.

ADASYN, on the other hand, adapts the synthetic sample generation process based on the difficulty of classification for different instances. It generates more synthetic samples for those instances that are relatively harder to classify correctly. ADASYN aims to improve the effectiveness of SMOTE in handling imbalanced datasets by focusing on areas of the feature space where the minority class is more challenging to predict.

By applying both SMOTE and ADASYN analyses, the team was able to create additional synthetic samples for the minority class, thereby increasing its representation in the dataset. This helped to mitigate the class imbalance issue and provided the machine learning model with a more balanced dataset for training. Ultimately, this approach aimed to improve the accuracy and performance of the model in detecting fake reviews, contributing to the team's goal of developing a highly accurate machine for reliable review detection.

Future Work

Future directions for this project encompass a range of potential improvements and refinements. These may include, among other strategies, the exploration of alternative algorithms, as well as the systematic adjustment of hyperparameters such as epoch and batch sizes. Additionally, Incorporating a more extensive and diverse range of data points may further enhance the performance and generalizability of the developed models.

Team Member Contributions:

Team Member Names	Team Member Contributions
Ethan Vaz Falcão	Contributed significantly to task 1 and task 2 of the project. For task 2, I worked on the methods and code and researched feature engineering methods. I also assisted with task 1 and used my PC to run both task 1 and task 2 fully. Additionally, I helped with the report paper and the slides.
Emre Sabaz	I worked on improving the task 1 methods and code through research. However, due to the limitations of the PC RAM, I was unable to fully run the code. In addition to this, I also contributed to the report paper and slides.
Maanav Iyengar	Set up meetings. Researched and assisted in the methods for tasks 1 and 2. Worked on the report and presentation.
Nur Fateemah	Contributed significantly to task 1, For task 1 experimented with different models and preprocessing methods did research on task 2 and helped with task 2, completed analysis of results report paper, and helped with the slides.
Sarah Kogan	Helped with slides and presentation