FAKE REVIEW DETECTION SYSTEM

Aayush Kapoor | Aditya Jain | Shashank Shekhar Singh | Vaibhav Wali | Vasu Khanna | Yatish Garg

Problem Statement

The issue being studied in this research project is the growing phenomenon of phoney reviews on online review sites, which can have a detrimental effect on consumer choice and undermine the authority of user-generated material. Through the use of Information Retrieval, machine learning algorithms and natural language processing techniques, this study seeks to provide an effective method for identifying false internet reviews. By extracting additional data including sentiment, confirmed purchases, ratings, product category, and total score, the suggested approach seeks to differentiate between real and false reviews.This project aims to provide a trustworthy and accurate method to address the issue of opinion spamming in online platforms by utilising a variety of classifiers such as Naive Bayes, SVM, Random Forest, Decision Trees, and Logistic Regression, as well as advanced natural language processing techniques such as LSTMs, transformers, and the BERT model. The usefulness of the suggested solution is assessed using a number of measures, including accuracy, precision, recall, and F1 score, to show that it is capable of identifying bogus reviews and offering users reliable user-generated information.

Motivation

Consumers now largely rely on online reviews to help them make educated purchasing decisions, and they have become an essential component of the decision-making process. However, the prevalence of phoney reviews has raised concerns about the veracity of user-generated \ material, which has various detrimental effects on both customers and businesses. False reviews are frequently found online, which has the ability to mislead

customers, harm a company's brand, and ultimately lower sales. Therefore, it is crucial to create a technique that can identify bogus reviews and eliminate them from internet platforms.

This research study proposes a method for identifying bogus reviews using a mix of machine learning algorithms and natural language processing techniques in order to solve the issue of opinion spamming. The suggested approach extracts additional characteristics that offer details about the reliability of reviews using the BERT model and information retrieval techniques. The suggested solution tries to achieve high accuracy in identifying fraudulent reviews by utilising a variety of classification algorithms and cutting-edge NLP methods.

Methodology

Our proposed solution for detecting fake reviews combined machine learning algorithms and natural language processing techniques. The methodology involved the following steps:

Extracting additional features:We scrapped two datasets as we could not find a single dataset with all the required features. We extracted features such as sentiment of the review, helpfulness score, ratings, product category, and overall score to gain insights into the authenticity of the reviews. By comparing these features between genuine and fake reviews, we identified patterns that distinguish between the two types of reviews.

Classification algorithms: We used Naive Bayes, SVM, Random Forest, Decision Trees, and Logistic Regression classifiers to perform the classification. These classifiers were trained on the preprocessed data and the additional features extracted earlier. We used techniques like cross-validation and hyperparameter tuning to improve the accuracy of the models.

Advanced natural language processing techniques: We also used advanced techniques like LSTMs, transformers, BERT (Bidirectional Encoder Representations from Transformers) model, and recurrent neural networks to capture the context and semantics of the text data and provide more accurate predictions.

Model evaluation: We evaluated the performance of the models using metrics such as accuracy, precision, recall, and F1 score. We also performed a comparative analysis of the different models to identify the best-performing model for detecting fake reviews.

The proposed methodology aimed to achieve high accuracy in detecting fake reviews by using a combination of features and classifiers. By using machine learning and natural language processing techniques, we provided a reliable solution for addressing the problem of opinion spamming in online websites. Specifically, we aimed to train a precise cumulative model by using both LSTM and BERT, which could provide more accurate predictions than traditional machine learning models. Overall, the proposed method offered an effective approach to detecting fake reviews and providing trustworthy user-generated content for consumers.

Related Works

Since 2007, the study of fake review detection has been conducted through review spamming analysis . The case of Amazon was looked at in this study, and the authors came to the conclusion that manually labelling fake reviews can be difficult because fake reviewers may carefully craft their reviews to make them more trustworthy for other users. As a result, they suggested using duplicates or nearly duplicates as spam to create a model that could identify fake reviews . Additionally, research on distributional footprints has demonstrated a link between distribution anomalies and deceptive hotel and Amazon product reviews. Some of the links we have referred from are :

https://ieeexplore.ieee.org/document/8335018
→This paper proposes a CNN-based approach for fake review detection and compares its performance with traditional machine learning methods.

Fake Review Detection: Classification and Analysis of Real and Pseudo Reviews →This paper describes features used in the model including sentiment analysis, part - of - speech tagging. And user behaviour analysis. The author concludes with the approach that it can automatically detect and remove fake reviews, improving the overall quality.

Detection of fake reviews using NLP &Sentiment Analysis | IEEE Conference Publication→This paper uses CNN and DT. This system extracts features from the text of reviews using CNN and DT to classify whether they are genuine or fake.

Detecting Fake Reviews Utilizing Semantic and Emotion Model | IEEE Conference Publication → This paper proposes a heuristic-based approach for detecting opinion spam. The
approach uses the sentiment of the review and the frequency of specific words to identify spam.

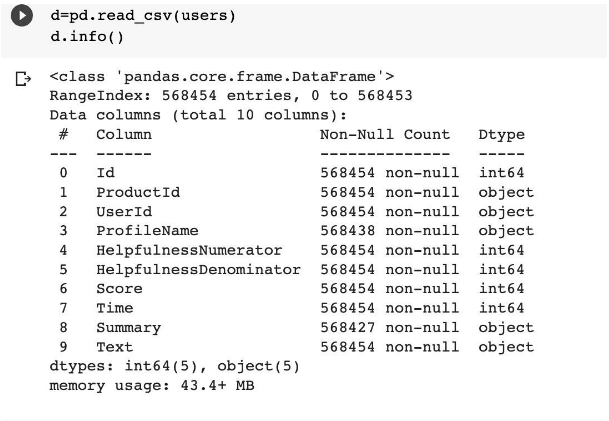
A Study on Identification of Important Features for Efficient Detection of Fake Reviews | IEEE Conference Publication → This paper discusses the impact of the fake review on business, and the different techniques. Used to detect them, including ML, NLP, DL and data mining.

A Deep Learning Approach for Fake Review Detection →This paper proposes a deep learning framework that combines CNN and ANN for fake review detection and achieves high accuracy on a large dataset.

Detecting Fake Reviews Using Convolutional Neural Networks" → by H. F. El-Sofany et al. This paper presents a CNN-based approach for fake review detection and compares its performance with traditional machine learning methods on a dataset of Amazon product reviews.

Dataset Description

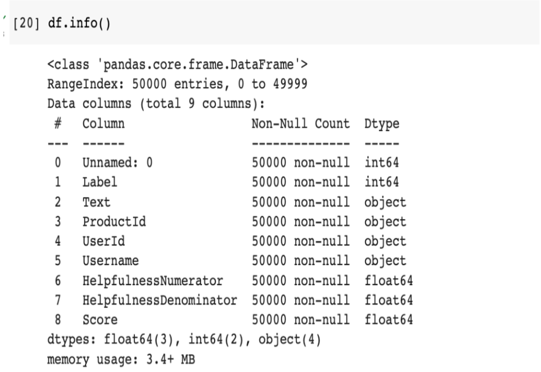
Initial dataset:



```
d=pd.read_csv(users)
d.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 568454 entries, 0 to 568453
Data columns (total 10 columns):
 #   Column                  Non-Null Count   Dtype 
---  ------                  --------------   ----- 
 0   Id                      568454 non-null  int64 
 1   ProductId               568454 non-null  object
 2   UserId                  568454 non-null  object
 3   ProfileName             568438 non-null  object
 4   HelpfulnessNumerator    568454 non-null  int64 
 5   HelpfulnessDenominator  568454 non-null  int64 
 6   Score                   568454 non-null  int64 
 7   Time                    568454 non-null  int64 
 8   Summary                 568427 non-null  object
 9   Text                    568454 non-null  object
dtypes: int64(5), object(5)
memory usage: 43.4+ MB
```

We removed some of the columns such as 'Id'and 'time' from our dataset because they were irrelevant for our model. Below are the final columns in our data set.

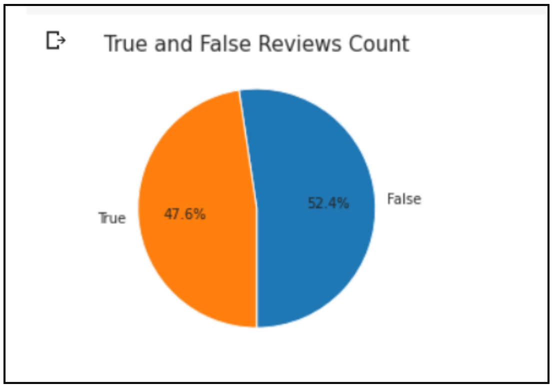


```
[20] df.info()
```

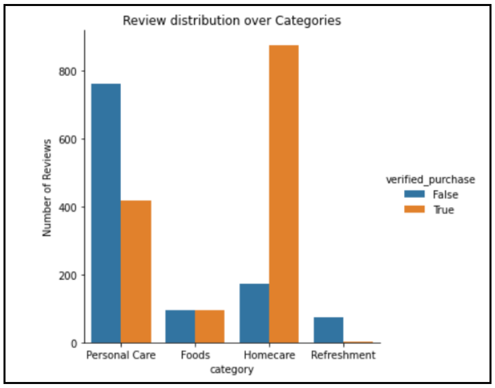
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50000 entries, 0 to 49999
Data columns (total 9 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Unnamed: 0              50000 non-null  int64  
 1   Label                   50000 non-null  int64  
 2   Text                    50000 non-null  object 
 3   ProductId               50000 non-null  object 
 4   UserId                  50000 non-null  object 
 5   Username                50000 non-null  object 
 6   HelpfulnessNumerator    50000 non-null  float64
 7   HelpfulnessDenominator  50000 non-null  float64
 8   Score                   50000 non-null  float64
dtypes: float64(3), int64(2), object(4)
memory usage: 3.4+ MB
```

The dataset taken for our model is the Amazon Reviews dataset. It is focused on the products sold on Amazon website and the reviews that are added by the customers about the products. The dataset contains 2500 rows and 32 columns.

Exploratory Data Analysis



Percentage of True and False Reviews in the dataset



No of True and False Reviews per category

review collected
part promotion
great
collected part
love
bottle
really
scent
one
face
time
soft
use
clothes
wipe
sensitive
dry
clean
always
price
nice
fresh
hair
good
lovely
make
well
wash
feel
fragrance
bought
washing
day
amazing
brand
buy
used
using
say
happy
need
tried
year
value money
little
much
quite
best
gentle
keep
simple
shower gel
perfect
buying

WordCloud showing common words in the reviews

Data Preprocessing

Few columns contain many null values, so replacing these values with the mean values of the columns.
Duplicate values were also removed.
Feature Transformation: The data present is categorical, so the string values have been scaled to Integer for model prediction.
Feature selection: On the basis of Information Gain, some of the attributes like dimensions, time-of-publication, category and few more columns which are product descriptive and which provide low information gain for review (dependent variable) are dropped.

After these steps, text processing was done like
spelling correction: using TextBlob()
removing punctuations: using regex,
removing stopwords: from NLTK SW library ,
lemmatization and tokenization.

Model Training

After preprocessing the data, we have used several Machine Learning models to classify reviews based on the features in the sample.
We have used the following classifiers:

1. Logistic Regression: It is a type of regression used in case of classification problems. It learns a linear relationship from the given dataset and then introduces a non-linearity in the form of the Sigmoid function.

2. Gaussian Naive Bayes: It is a type of classification model which uses Bayes algorithm. It is easy and fast in multiclass classification as it needs less training data. It is used to determine the benchmark performance of the models.

3. Random Forests Classifier: It is ensemble learning of Decision Trees(which provides interpretability and is non-parametric in nature) where some weak classifiers are combined and the prediction is done by majority voting for classification problems.

4. Decision Tree Classifier: A decision tree is a non-parametric supervised learning algorithm which provides interpretability while doing classification. At each level, a feature is chosen as per its information gain or entropy for classifying data and final classification is obtained at the leaf level.

5. SVM : A support vector machine (SVM) is a supervised learning algorithm to classify or

predict data groups. The goal of the SVM is to determine the unique decision boundary known as Optimum Separating Hyperplane (OSH) that can segregate n-dimensional space into the required number of regions for classification.

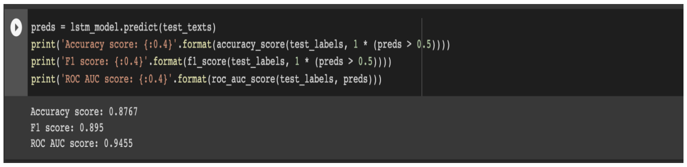
6. Deep Learning Models used -
CNN - Convolutional Neural Networks are deep learning models used to reduce multidimensional data to scalar data so that it can be used in a neural network for classification.
ANN- Artificial Neural Networks are based on the working of a human brain neuron. Each node in the network is assigned a weight to it which is updated during backpropagation of the error between predicted and actual labels. Finally we get the set of optimal weights.
LSTM - LSTM is a type of recurrent neural network that uses a memory cell and gating mechanisms to selectively control the flow of information through the network, allowing it to maintain long-term dependencies in sequential data.
BERT - BERT is a pre-trained deep learning model based on the transformer architecture, designed to understand natural language by training on large amounts of text data using bidirectional training. BERT can be fine-tuned on a variety of natural language processing tasks to achieve state-of-the-art results.

Results and Analysis

We have come to the conclusion that BERT and LSTM perform the best on our data, we report accuracies of 88% and 90% on the testing dataset respectively. The comparison table with SOTA is given below -

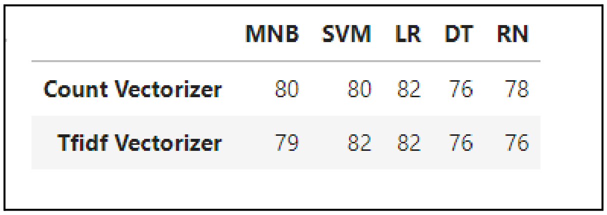
	F-1	Accuracy	ROC-AUC
SOTA	0.965	0.988	0.994
LSTM	0.895	0.8767	0.9455
BERT	0.912	0.8918	0.962

Accuracy of LSTM



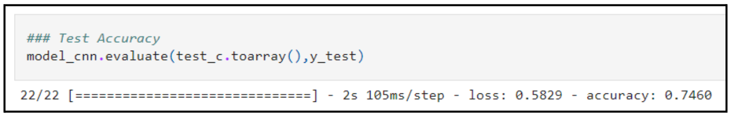
```
preds = lstm_model.predict(test_texts)
print('Accuracy score: {:0.4}'.format(accuracy_score(test_labels, 1 * (preds > 0.5))))
print('F1 score: {:0.4}'.format(f1_score(test_labels, 1 * (preds > 0.5))))
print('ROC AUC score: {:0.4}'.format(roc_auc_score(test_labels, preds)))

Accuracy score: 0.8767
F1 score: 0.895
ROC AUC score: 0.9455
```

Accuracies of baseline models



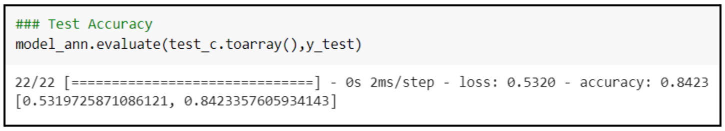
	MNB	SVM	LR	DT	RN
Count Vectorizer	80	80	82	76	78
Tfidf Vectorizer	79	82	82	76	76

Accuracy of CNN



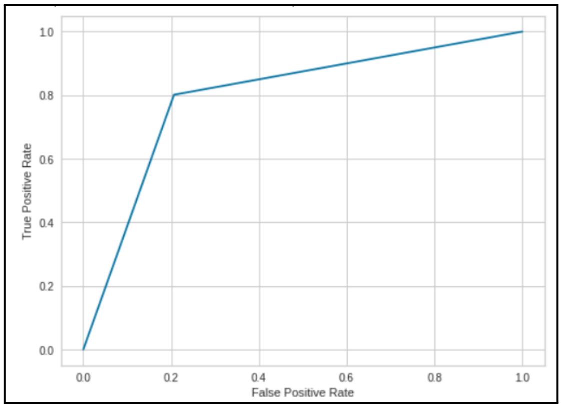
```
### Test Accuracy
model_cnn.evaluate(test_c.toarray(),y_test)

22/22 [==============================] - 2s 105ms/step - loss: 0.5829 - accuracy: 0.7460
```

Accuracy of ANN

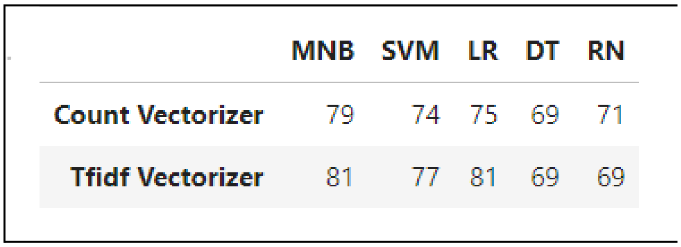


```
### Test Accuracy
model_ann.evaluate(test_c.toarray(),y_test)

22/22 [==============================] - 0s 2ms/step - loss: 0.5320 - accuracy: 0.8423
[0.5319725871086121, 0.8423357605934143]
```

ROC- Curve of CNN

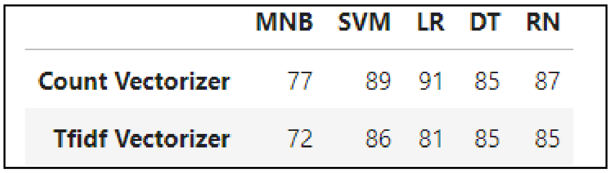


Precisions of various models



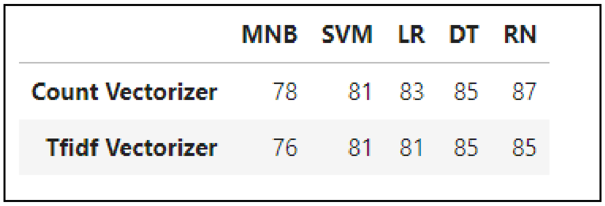
	MNB	SVM	LR	DT	RN
Count Vectorizer	79	74	75	69	71
Tfidf Vectorizer	81	77	81	69	69

Recalls of various models



	MNB	SVM	LR	DT	RN
Count Vectorizer	77	89	91	85	87
Tfidf Vectorizer	72	86	81	85	85

F1-scores of various models



	MNB	SVM	LR	DT	RN
Count Vectorizer	78	81	83	85	87
Tfidf Vectorizer	76	81	81	85	85

From the above results , the overall accuracy was close to 80% for all the models. Also the precision is higher for some models while recall The CNN model was trained on a dataset using binary cross entropy loss. Adam optimizer. The model achieved 74.6% accuracy on the test data. The confusion matrix plotted shows the true positives , true negatives , false positives and false negatives. The area under ROC came out as 0.797,which is an indication of the model's ability to distinguish between positive and negative classes.

In ANN the model has achieved the accuracy of 84.23% on the test set, and the area under the CURVE (ROC) is 0.82. The ROC curve is a graphical representation of the performance of a binary classification model at different

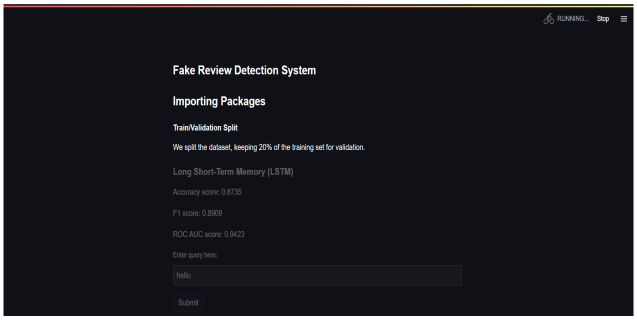
The LSTM based Model had an accuracy of 87.67 whereas the BERT model had an accuracy score of 89.19, the ROC AUC score was 94.55 and 96.2 respectively. These outperformed all the other ML based techniques and pushed our model in the realm of State of the art model with AUC ROC value of ~99.4. The Higher the AUC, the better the model's performance.

Novelty

We have implemented a small recommendation system which recommends the user products based on the user's choices. There are 2 types of recommendation systems in our project one is a general recommendation system and the other one is a personalised recommendation system, these have been made with the use of collaborative filtering. We have also shown the top 10 and the worst 10 users on the basis of their helpfulness in the reviews.

Front End Design

Our model takes user reviews and classifies the input into true or false reviews using ANN classifier. Some screenshots of our model are:



Fake Review Detection System

Importing Packages

Train/Validation Split

We split the dataset, keeping 20% of the training set for validation.

Long Short-Term Memory (LSTM)

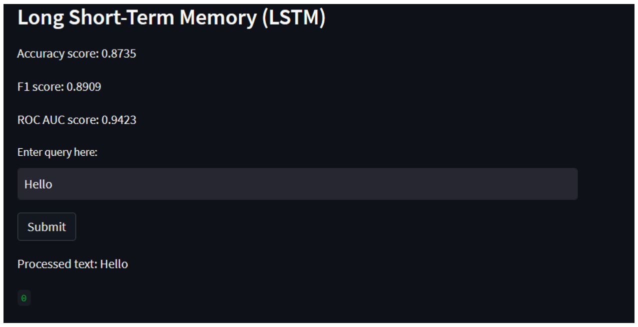
Accuracy score: 0.8735

F1 score: 0.8909

ROC AUC score: 0.9423

Enter query here:

hello



Long Short-Term Memory (LSTM)

Accuracy score: 0.8735

F1 score: 0.8909

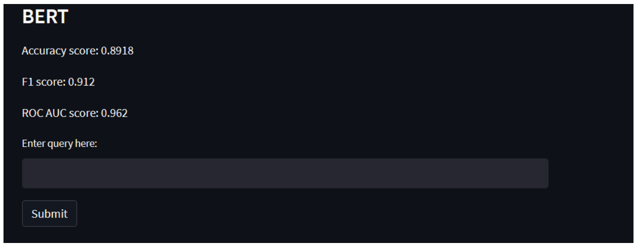
ROC AUC score: 0.9423

Enter query here:

Hello

Processed text: Hello

```
0
```



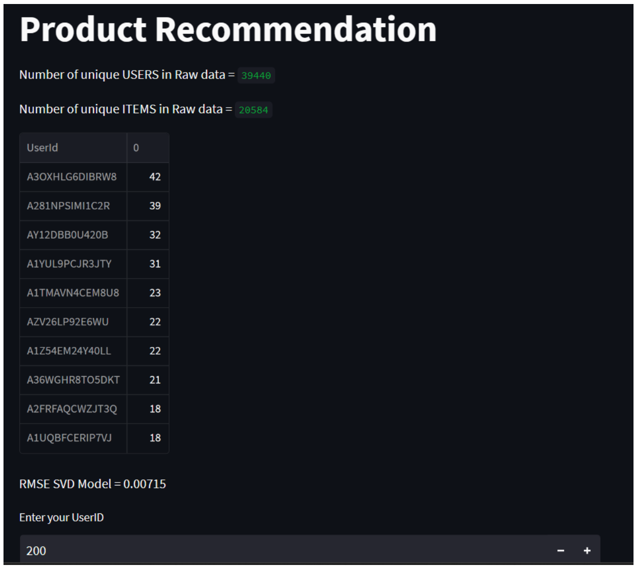
BERT

Accuracy score: 0.8918

F1 score: 0.912

ROC AUC score: 0.962

Enter query here:



Product Recommendation

Number of unique USERS in Raw data = `39440`

Number of unique ITEMS in Raw data = `20584`

UserId	0
A3OXHLG6DIBRW8	42
A281NPSIMI1C2R	39
AY12DBB0U420B	32
A1YUL9PCJR3JTY	31
A1TMAVN4CEM8U8	23
AZV26LP92E6WU	22
A1Z54EM24Y40LL	22
A36WGHR8TO5DKT	21
A2FRFAQCWZJT3Q	18
A1UQBFCERIP7VJ	18

RMSE SVD Model = 0.00715

Enter your UserID

200

Conclusion

In conclusion, the goal of detecting fake reviews is to weed out the biassed ones and ensure that customers can make informed decisions about products. Our study used machine learning algorithms, including Random Forests, LSTM, and BERT, to classify our dataset. Our results showed that both LSTM(88% accuracy) and BERT(89% accuracy) outperformed the other classifiers, highlighting the effectiveness of absolute classifiers in this task. Furthermore, data visualisation aided in exploring the dataset and identifying key features that improved classification accuracy. The high accuracy achieved by the various algorithms underscores their suitability for this task.

Ultimately, our approach empowers users to identify the most trustworthy reviews and make informed purchasing decisions.

Future Work

There are several avenues for future research that could enhance the accuracy and efficiency of fake review detection systems.

Developing a method for unsupervised learning of unlabeled data could improve the ability to identify fake reviews. This would involve training a model on a large dataset of reviews without any prior knowledge of which ones are genuine or fake. The model could then be used to identify patterns and anomalies in new reviews, providing a more comprehensive and robust fake review detection system.

Overall, these future works have the potential to enhance the accuracy and efficiency of fake review detection systems, ultimately benefiting both customers and businesses by promoting trust and transparency in online reviews.

Contributions:

EDA- Shashank and Vasu
Baseline models- Aditya and Aayush
Mid Evaluation Models- Yatish and Vaibhav
Data Scraping- Vasu and Shashank
Novelty - Vaibhav and Aayush
Front End- Yatish
Final Models - Combined effort
Report Making - Combined effort
PPT - Combined effort