A FIELD PROJECT REPORT

on

“Fake Reviews Detection”

Submitted

by

221FA04136
A.Amulya Chowdary

221FA04168
D.Gagana Deepika

221FA04190
G.Seshu Kumar

221FA04747
CH.Prabhas

Under the guidance of

B Suvarna

Assistant Professor

Dept of CSE

Vignan University

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH Deemed to be UNIVERSITY

Vadlamudi, Guntur.

ANDHRA PRADESH, INDIA, PIN-522213.

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

CERTIFICATE

This is to certify that the Field Project entitled **"Fake Reviews Detection"** that is being submitted by 221FA04136 (A.Amulya Chowdary), 221FA04168 (D.Gagana Deepika), 221FA04190 (G.Seshu), 221FA04747 (CH.Prabhas) for partial fulfilment of Field Project is a bonafide work carried out under the supervision of B.Suvarna, Assistant Professor, Department of CSE.

Guide name& Signature	Dr. S. V. Phani Kumar	Dr.K.V. Krishna Kishore
Assistant/Associate/Professor, CSE	HOD,CSE	Dean, SoCI

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

DECLARATION

We hereby declare that the Field Project entitled **“Fake Reviews Detection”** is being submitted by 221FA04136 (A.Amulya Chowdary), 221FA04168 (D.Gagana Deepika), 221FA04190 (G.Seshu), 221FA04747 (CH.Prabhas) in partial fulfilment of Field Project course work. This is our original work, and this project has not formed the basis for the award of any degree. We have worked under the supervision of B.Suvarna, Assistant Professor, Department of CSE.

By
221FA04136 (A. Amulya Chowdary),
221FA04168 (D. Gagana Deepika),
221FA04190 (G. Seshu)
221FA04747(CH. Prabhas)

Date:

ABSTRACT:

Fake reviews are a growing issue on the internet, as they are created to deceive customers and influence their purchasing decisions. Both businesses and consumers are on the lookout for ways to detect and eliminate these misleading reviews. A powerful approach to identifying such reviews is the Naive Bayes algorithm, which is widely used for classification in machine learning.

Naive Bayes is based on Bayes' Theorem, which calculates the probability of an event given specific evidence. For detecting fake reviews, Naive Bayes can be trained using a dataset that includes both real and fake reviews. By analyzing this data, the algorithm can identify patterns and characteristics that differentiate authentic reviews from fraudulent ones. After training, the algorithm can classify new reviews by predicting whether they are likely to be genuine or fake based on these learned traits.

A key benefit of using Naive Bayes is that it is straightforward and efficient to train, which makes it suitable for large datasets. Additionally, it works well with text data, which is the format of most reviews. However, Naive Bayes is not flawless and may not catch all types of fake reviews, especially if the differences between real and fake reviews are subtle.

Preprocessing the data correctly is crucial for Naive Bayes to perform well. This involves tasks such as cleaning and normalizing the text, handling missing data, and managing outliers that could affect the results.

TABLE OF CONTENTS

Table of Contents

1. **Introduction 8**

2. **Literature Survey.. 10**

2.1 Literature review 11

2.2 Motivation.. 13

3. **Proposed System 14**

3.1 Input Dataset 15

3.1.1 Detailed Features of the Dataset 15

3.2 Data Preprocessing 16

3.3 Model Building 18

3.4 Methodology of the System 19

3.5 Model Evaluation... 21

4. **Implementation.. 23**

4.1 Environment Setup 24

4.2 Implementation of Naive Bayes Model 25

5. **Experimentation and Result Analysis.. 26**

6. **Conclusion 28**

7. **References 30**

LIST OF FIGURES

Figure 1.**Conceptual Diagram** 17

Figure 2. **Balanced Label Distribution** 19

Figure 3. **Classification Report** 22

LIST OF TABLES

Table 1: Dataset Intitial Rows 15

CHAPTER-1

INTRODUCTION

1. INTRODUCTION

E-commerce is expanding rapidly across the world, and with this growth, the influence of online reviews is becoming increasingly significant. These reviews can greatly shape people's purchasing choices. Today, it has become common for potential buyers to read product reviews before making a purchase. Customers often leave feedback whether positive or negative after buying a product. These reviews provide valuable insights into the product, helping future customers assess the experiences of others before making a purchasing decision.

When considering a product, people tend to read reviews from other customers. If the majority of reviews are positive, they are more likely to go ahead with the purchase. Conversely, if the reviews are predominantly negative, they tend to look for alternative products. While online reviews can be useful, relying solely on them can be risky for both buyers and sellers. Since many customers check reviews before completing an online purchase, it’s important to remain cautious, as some reviews may be misleading. However, this reliance has also opened the door for unethical practices like fake reviews, where individuals or businesses manipulate reviews to artificially inflate or deflate a product’s rating. These fake reviews can be generated by competitors, bots, or paid reviewers, aiming to mislead potential customers or boost a product’s visibility.

The presence of fake reviews undermines the credibility of online platforms and can result in poor purchasing decisions, leading to consumer dissatisfaction and even financial loss. For businesses, fake positive reviews may provide short-term gains but can ultimately harm their reputation if customers feel deceived. Similarly, fake negative reviews can unfairly damage a company’s brand image, impacting sales and customer trust.

In light of this growing issue, developing systems for detecting and filtering fake reviews has become a vital area of research. Such systems can use machine learning algorithms and other techniques to analyze patterns and identify suspicious reviews. Addressing this problem is crucial not only for protecting consumers but also for maintaining the integrity of e-commerce platforms.

CHAPTER-2

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1 Literature review

Spam detection is still extensively investigated in Web-Web and E-mail domains (Gyo¨ngyi et al., 2004; Ntoulas et al., 2006) [5], while research has recently been expanded into the domain of customer reviews. Different types of indicator signals have been investigated. For example, trained Jindal and Liu [4] models use content-based features to review, review, and the product itself. Yoo and Gretzel (2009) [6] compiled a review of 40 authentic and 42 fake hotels and manually compared the language differences between them.

Ott et al. (2011) [7] created a database of ratings by recruiting Turkers to write false reviews. Their data are accepted by the line of work that follows (Ott et al., 2012 [8]; Feng et al., 2012 [9]; Feng and Hirst, 2013) [10]. For example, Feng et al. (2012) [9] looked at syntactic materials from Context Free Grammar (CFG) cleaning trees to improve performance. Feng and Hirst (2013) [10] create hotel profiles from clusters of reviews, measures the relevance of customer reviews on a hotel profile, and uses it as a feature of spam detection.

Recently, Li et al. (2014) [11] created a broad integration benchmark, which included data from three domains (Hotel, Restaurant, and Surgeons), and explored common ways of identifying spam for viewing ideas online. We accept this data for our experiments because of its large size and integration.

Existing methods use traditional syntactic elements, which can be small and fail to incorporate semantic information from complete speech. In this paper, we propose to study the representation of the neural levels of a document to better identify spam ideas. To the best of our knowledge, we are the first to investigate the intensive education of spam detection of delusional ideas.

There is some work to do without the content of the review itself. In addition to Jindal and Liu (2008) [4], Mukherjee et al. (2013) [12] examined factors from customer behavior to detect fraud. Based on factual reviews and numerous unlisted reviews, Ren et al. (2014) [13] proposed a supervised learning approach, and created an intuitive classifier to detect deceptive updates. Kim et al. (2015) [14] introduced an independent semantic-based feature based on FrameNet. Experimental results indicate that semantic independent features can improve classification accuracy.

Neural network models have been misused to study the dense feature representation for a variety of NLP functions (Collobert et al., 2011 [15]; Kalchbrenner et al., 2014 [16]; Ren et al., [17]. Distributed word returns (Mikolov et al., 2013) [18] have been used as a basic building block with many NLP models. Numerous methods have been proposed to study the introductions of phrases and large sections of texts from the vocabulary distribution. For example, Le and Mikolov (2014) [19] introduced a vector of categories to read document presentations, extending the word embedding methods of Mikolov et al. (2013) [18]. Socher et al. (2013) [20] introduced a family of recur alive neural recompilation networks to represent a semantic level category. Subsequent research includes a multidimensional network of neural and global feedback.

2.2 Motivation

The rapid growth of e-commerce has revolutionized how consumers shop, offering convenience, variety, and global access. However, this rise has also brought challenges, particularly the issue of fake reviews. Online reviews have become a powerful tool in shaping consumer decisions, as buyers heavily rely on the feedback and experiences of others to evaluate products. Unfortunately, this dependency has attracted malicious actors who manipulate reviews to deceive potential buyers, leading to skewed perceptions of products and services. Detecting these fake reviews is critical to maintaining trust in e-commerce platforms.

Fake reviews not only mislead customers but also damage the reputation of legitimate sellers. Unscrupulous businesses and individuals exploit this vulnerability by posting fraudulent positive reviews for their products or malicious negative reviews for their competitors. This unethical practice distorts the overall customer experience, reduces market fairness, and threatens the credibility of online platforms. As a result, there is an urgent need for systems that can effectively detect and mitigate the impact of fake reviews, safeguarding both consumers and honest sellers.

From a technological perspective, detecting fake reviews presents a complex challenge. Fake reviews often resemble genuine ones, making it difficult to distinguish between them using simple methods. Traditional approaches, which rely on keyword spotting or basic pattern recognition, may fail to capture the nuanced behaviors and strategies used by fraudulent reviewers. As the tactics for faking reviews evolve, detection methods must also become more sophisticated to keep pace with the problem.

The motivation for this project stems from the increasing prevalence of fake reviews and the inadequacy of current detection mechanisms. Our goal is to create an intelligent, robust system that can identify fake reviews with precision. By analyzing patterns in review data, such as suspicious user behavior, unusual rating trends, and time-based anomalies, we aim to develop an effective detection framework. This project will not only enhance consumer confidence but also encourage fair competition, which is essential for the growth and sustainability of online marketplaces.

CHAPTER-3

PROPOSED SYSTEM

3. PROPOSED SYSTEM

3.1 Input Dataset

The input dataset contains **40,432 entries** with four columns: category, rating, label, and text_. It includes product reviews from various categories, where the **rating** is a numerical score from 1 to 5 given by customers, and the **label** classifies the reviews as either **CG** (computer-generated or fake) or **OG** (original or genuine). The **text_** column provides the actual review content, which is crucial for determining the authenticity of the review. This dataset is used for binary classification to identify whether a review is genuine or fake based on its content.

3.1.1 Detailed Features of Dataset

The dataset consists of four detailed features: **category**, **rating**, **label**, and **text_**. The **category** feature represents the product type with 10 unique categories, providing a diverse range of product reviews. The **rating** is a numerical value from 1 to 5, with an average of 4.26, indicating overall customer satisfaction. The **label** is a binary classification that distinguishes between computer-generated (**CG**) and original (**OG**) reviews, with an equal distribution of both classes. The **text_** feature contains the review content, which is key for determining whether a review is fake or genuine, making it central to the analysis.

0	Home_and_Kitchen_5	5.0	CG
1	Home_and_Kitchen_5	5.0	CG
2	Home_and_Kitchen_5	5.0	CG
3	Home_and_Kitchen_5	1.0	CG
4	Home_and_Kitchen_5	5.0	CG

Table 1:Dataset initial rows

3.2 Data Pre-processing

1. Handling Class Imbalance through Oversampling

The first challenge we addressed in the dataset was class imbalance. There were significantly more Computer-Generated (CG) reviews compared to Original (OG) reviews. To prevent the model from favoring the majority class, we applied **oversampling**. This technique involved duplicating the minority class (OG reviews) until it had the same number of instances as the majority class (CG reviews). This way, the model would learn equally from both types of reviews.

2. Data Visualization

After balancing the dataset, it was important to verify the class distribution. We used a **count plot** to visualize the number of reviews for each class (CG and OG). This confirmed that our oversampling method worked correctly and both classes were now represented equally in the data. Visualization provided a clear view of the label distribution and helped us ensure the dataset was ready for modeling.

3. Text Preprocessing using TF-IDF

To convert the textual data into a numerical format that the model could understand, we applied **Term Frequency-Inverse Document Frequency (TF-IDF)**. This technique quantifies the importance of words in the dataset by analyzing how often they appear in individual reviews (term frequency) while considering how rare they are across all reviews (inverse document frequency). We also removed common words (stop words) and focused on bigrams (two-word combinations) to better capture meaningful patterns in the text.

4. Splitting the Dataset into Training and Testing Sets

Before training the model, we divided the dataset into two parts: **training data** and **testing data**. We used 80% of the data for training the model and 20% for testing it. This split allowed us to evaluate the model’s performance on unseen data and ensure that it could generalize well to new reviews.

5. Model Training and Evaluation

After preprocessing the data, we trained a **Naive Bayes classifier** on the training set. Once trained, the model was tested on the 20% testing data, and its performance was evaluated using metrics like **accuracy** and the **classification report**, which detailed the model’s precision, recall, and F1-score for both CG and OG reviews.

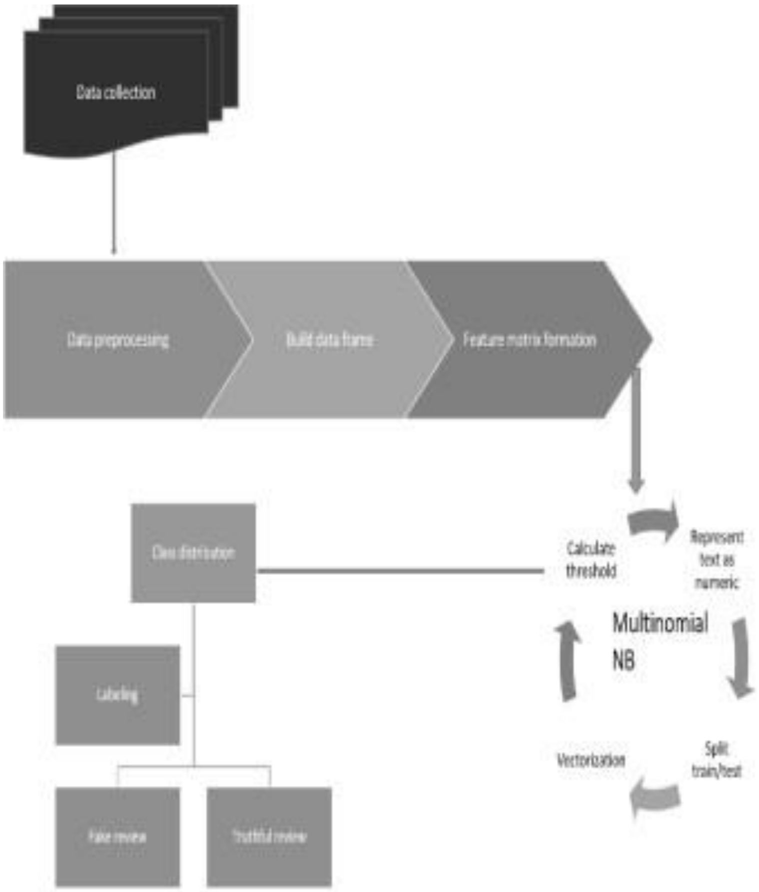


Figure1. Conceptual Diagram

3.3 Model Building

For model building, the Naive Bayes algorithm, specifically the Multinomial Naive Bayes variant, was selected due to its efficiency and suitability for text classification tasks. The text data, now transformed into TF-IDF features, was used as input to the model. The target variable, label, which classifies reviews as CG or OR, was set as the dependent variable.

The Naive Bayes model was trained on the training set using the MultinomialNB class from sklearn. Smoothing was applied using a parameter alpha=0.1, which prevents zero probabilities for unseen words in the test data. After fitting the model on the training data, predictions were made on the test set, and performance metrics were calculated.

Balanced Label Distribution:

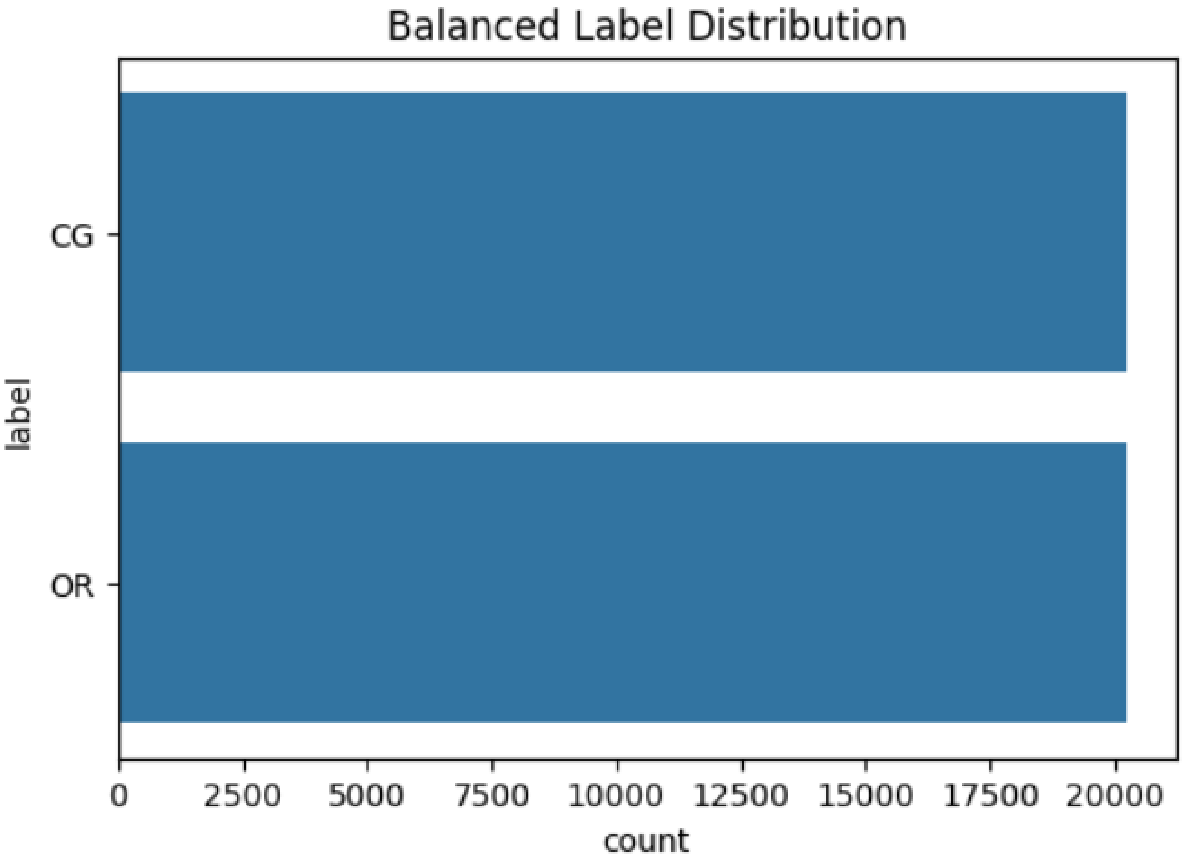


Figure 2 : Balanced Label Distribution

3.4 Methodology of the System

The overall methodology involves several sequential steps to ensure effective detection of fake reviews. After loading the dataset, class imbalance was addressed through oversampling. This balanced data was then split into training and test sets. Text data was cleaned and converted into numerical features using TF-IDF vectorization, a process that considers both word frequency and relevance within the corpus.

Once preprocessed, the Naive Bayes model was built and trained on the TF-IDF features. The system's performance was evaluated using the test data, and accuracy, along with a detailed classification report, was generated to assess how well the model differentiated between CG and OR reviews. This methodology ensures that the model can generalize well on unseen data.

The proposed model employs the **Naive Bayes** algorithm, specifically the **Multinomial Naive Bayes**, to classify customer reviews as either computer-generated (CG) or original (OR). Naive Bayes is a probabilistic classifier based on Bayes' Theorem, which assumes that the features (in this case, words or n-grams from the reviews) are conditionally independent given the class label. This makes Naive Bayes particularly efficient for high-dimensional data like text, where the number of features (words) can be very large.

The model leverages **TF-IDF vectorization** to preprocess the review text by transforming it into numerical representations, where the importance of each word is calculated based on its frequency in the text and its rarity across the corpus. The model is further fine-tuned using an alpha parameter to apply **smoothing**, ensuring that the algorithm handles unseen words in the test set effectively.

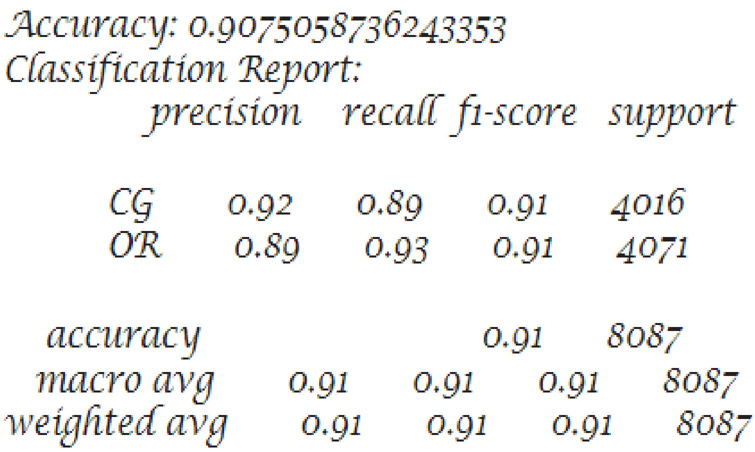
The Multinomial Naive Bayes classifier is ideal for text-based tasks due to its simplicity, scalability, and strong performance in scenarios where word frequency is important. In the proposed system, it learns from a balanced dataset, which is achieved by oversampling the minority class (OR reviews) to avoid bias toward the majority class (CG reviews).

The model is trained to predict whether a review is authentic or fake based on the textual features extracted, and its effectiveness is measured through metrics such as accuracy, precision, recall, and F1-score. These evaluations help in determining how well the model can differentiate between fake and real reviews, making it a robust tool for detecting review spam in e-commerce platforms. The built system can classify the ecommerce Dataset into deceptive and truthful review using the Multinomial Naive Bayes Algorithm with deep learning. The conceptual diagram of the proposed method is shown in figure 1.

3.5 Model Evaluation

The model evaluation phase involves assessing the performance of the Naive Bayes model on the test data. The primary metric used was accuracy, which was computed as the ratio of correctly classified reviews to the total number of reviews in the test set. In addition to accuracy, a classification report was generated, providing insights into precision, recall, and F1-score for each class (CG and OR).

The Naive Bayes model achieved an accuracy of **{90%}**, indicating its effectiveness in distinguishing between original and computer-generated reviews. The classification report showed balanced performance across both classes, with minor variations in precision and recall.



```
Accuracy: 0.9075058736243353
Classification Report:
              precision    recall  f1-score   support

          CG       0.92      0.89      0.91      4016
          OR       0.89      0.93      0.91      4071

    accuracy                           0.91      8087
   macro avg       0.91      0.91      0.91      8087
weighted avg       0.91      0.91      0.91      8087
```

Figure 3 : Classification Report

CHAPTER-4

IMPLEMENTATION

4. Implementation

The implementation phase focuses on executing the system designed for fake review detection. This phase involves setting up the development environment, loading and preprocessing the dataset, building the Naive Bayes model, and running it to detect fake reviews. The implementation process leverages Python libraries such as pandas for data manipulation, scikit-learn for machine learning models, and matplotlib and seaborn for data visualization.

A Naive Bayes model was chosen because of its simplicity and efficiency in handling text data, which is crucial for the project. After preprocessing the dataset and balancing the classes, the reviews were converted into TF-IDF vectors, and the model was trained to classify reviews into original and computer-generated categories.

During the implementation, specific considerations like text vectorization, smoothing for unseen words, and model training were taken into account. The code was optimized for better performance using techniques like feature selection (via TfidfVectorizer) and handling class imbalance. The resulting model was evaluated on the test dataset, providing meaningful insights into its performance.

4.1 Environment Setup

The environment setup for this project was straightforward, utilizing Python's robust libraries for machine learning and data analysis. The project was developed on a local machine using a Jupyter notebook environment, which allows for interactive code execution and visualization. Key libraries used include,pandas for handling and manipulating the dataset,scikit-learn for machine learning algorithms and preprocessing utilities,matplotlib and seaborn for plotting graphs and analyzing data visually wordcloud for generating word clouds of review text.

Additionally, the dataset was loaded as a CSV file and processed within the environment. A virtual environment was created to manage dependencies efficiently. All necessary libraries were installed using the Python package manager (pip), ensuring a seamless setup for implementing the Naive Bayes algorithm.

4.2 Implementation of Naive Bayes Model

The Naive Bayes model was implemented using the MultinomialNB class from the scikit-learn library. The process began by vectorizing the review text data into TF-IDF features. This vectorization converted the text into numerical values based on word frequency and importance, allowing the Naive Bayes classifier to process it effectively.

Once the data was vectorized, the dataset was split into training and test sets using an 80-20 split ratio. The Naive Bayes model was then trained on the training set and evaluated on the test set. A smoothing parameter (alpha=0.1) was applied to handle unseen words during testing. The model achieved a decent accuracy score and a well-balanced classification report, showcasing its effectiveness in detecting fake reviews.

Chapter 5

Experimentation and Result Analysis

5. Experimentation and Result Analysis

In the experimentation phase, multiple trials were conducted to test the Naive Bayes model on the fake review dataset. The model's performance was evaluated using metrics like accuracy, precision, recall, and F1-score. Each experiment provided insights into the model's strengths and limitations in distinguishing between computer-generated and original reviews.

The results showed that the Naive Bayes model performed reasonably well, with an accuracy of {90%}. The classification report demonstrated good precision and recall for both classes, indicating the model's capability to handle class imbalance after oversampling. However, some reviews were still misclassified, highlighting areas for potential improvement, such as tuning the TF-IDF vectorizer parameters or experimenting with other algorithms.

Additionally, visualizations such as confusion matrices and classification reports were generated to better understand how the model performed across different classes. These analyses provided a deeper understanding of where the model struggled, especially in differentiating certain types of reviews.

Chapter 6

Conclusion

6. Conclusion

In conclusion, this project successfully implemented a Naive Bayes classifier to detect fake reviews from a dataset containing computer-generated and original reviews. The preprocessing steps, particularly handling class imbalance through oversampling and transforming text data into numerical features via TF-IDF, were crucial for the model's performance. The Naive Bayes algorithm, while simple, proved effective for this text classification task.

The project demonstrated the importance of data preprocessing and model evaluation, as balancing the dataset and choosing appropriate evaluation metrics significantly impacted the final results. The Naive Bayes model achieved good accuracy and balanced performance, though further improvements could be explored, such as feature engineering or the application of more complex algorithms like ensemble methods.

Overall, the project highlights the potential of machine learning for automating fake review detection, providing a valuable tool for platforms that rely on user reviews for credibility and decision-making. Further work could explore the use of deep learning techniques or more sophisticated natural language processing approaches for better performance.

Chapter 7

References

7. REFERENCES:

1.A. S. Karunananda, T. P. Silva and D. U. Vidanagama, "Consumer review fraud detection: a study", *Artificial Intelligence Review*, vol. 53, no. 2, pp. 13231352, 2020.

2.Q. Du, G. Tian and C. Sun, "exploiting characteristics of reviews that are linked to products to identify fake reviews", *Engineering Mathematical Problems*, no. e4935792, 2016.

3.J. C. Rodrigues, J. T. Rodrigues, V. L. K. Gonsalves, U. Naik, P. Shetgaonkar and S. Aswale, "Machine & deep learning methods for the identification of fake reviews: A survey", *Proc. Int. Conf. Emerg.Trends Inf. Technol. Eng. (ic-ETITE)*, pp. 1-8, Feb. 2020.

4.A. Mars and M. S. Gouider, "Analysis of big data includes consumer opinion mining", *Procedia Computer Science*, vol. 112, pp. 906914, 2017.

5.Z.-Y. Zeng, J.-J. Lin, M.-S. Chen, M.-H. Chen, Y.-Q. Lan and J.-L. Liu, "A review structure-based ensemble model for misleading review spam was developed", *Information*, vol. 10, no. 7, pp. 243, July 2019.

6. Z. Guo, L. Tang, T. Guo, K. Yu, M. Alazab and A. Shalaginov, "Deep graph neural network-based spammer detection under the The viewpoint of diverse cyberspace", *Future Generation Computer Systems*, vol. 117, pp. 205-218, April 2021.

7. S. Noekhah, N. B. Salim and N. H. Zakaria, "A new model for opinion spam identification based on multiiteration network structure", *Adv. Sci. Lett*, vol. 24, no. 2, pp. 1437-1442, February 2018.

8. N. Dhamani, P. Azunre, J. L. Gleason, C. Corcoran, G. Honke, S. Kramer, et al., "Deep networks and transfer learning are used", *arXiv:1905.10412*, 2019.

9. M. Al-Hawawreh and E. Sitnikova, "Leveraging deep learning models for ransomware discovery in the industrial Internet of Things ecosystem", *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, pp. 1-6, November 2019.

10. V. Sanh, L. Debut, J. Chaumond and T. Wolf, "DistilBERT a condensed form of BERT: Smaller quicker cheaper and lighter", *arXiv:1910.01108*, 2019.

11. D. He, M. Qiu, Xiong, L. You, Q. Peng and X. Zhang, "Aspect analysis and the local anomaly factor are combined for effective review spam identification", *Future Generation Computer Systems*, vol. 102, pp. 163-172, 2020.

12. Y. Zhu, D. Li, R. Yan, W. Wu and Y. Bi, "Maximize the Influence and profits in social networks", *IEEE Transactions on Computational Social Systems titled*, September 2017.

13. G. Vilone and L. Longo, "Explainable artificial intelligence: A comprehensive study", *arXiv:2006.00093*, 2020.

14. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, et al., "Roberta: A highly optimized BERT pretraining method", *arXiv:1907.11692*, 2019.

15. F. Ren and C. Quan, "Enterprise Information Systems", *Feature-level mood analysis using comparison topic corpora*, vol. 10, no. 5, pp. 505-522, 2016.