Fake Review Detection

ABSTRACT: Detection of fake reviews out of a massive collection of reviews having various distinct categories like Home and Office, Sports, etc. with each review having a corresponding rating, label i.e. CG(Computer Generated Review) and OR(Original Review generated by humans) and the review text.Main task is to detect whether a given review is fraudulent or not. If it is computer generated, it is considered fake otherwise not.

INTRODUCTION: Now a days because of pandemic situation, it is observed that there is very fast increase in e-commerce. Society prefers e-banking, online shopping, etc. for their convenience. E-commerce allows customer to give feedback about the service. And the presence of these feedback can become source of information to another new customer. In case of online shopping user buys the product only by reading the reviews of the particular product. That means reviews are playing very important role in online shopping. But in this scenario, if the reviews about the product are fake then it will definitely give wrong conclusion about the product. We know that reviews are of two categories i.e. genuine and fake. Fake reviews can be good or bad. There are different types of fake reviews like if seller post any product for selling he himself ask to his social members to comment on that product or sometimes user himself/herself did not buy the product justcomment on it. so these type of reviews are fake.Different techniques like pre-processing, feature selection, tokenization, web scraping, etc. are used while developing this system.Different machine learning models are being compared here to choose the best, which turns out to be the support vector classifier which is approximately 88 percent. These labelled fake reviews can be further checked by other factors such as number of reported cases to verify the prediction of the ml model. Using this system user can differentiate the reviews of product in two categories i.e. fake or genuine. And only by reading genuine reviews list user first saves his/her time then get the accurate judgement about the product. And finally we proved the effectiveness of the system

LITERATURE REVIEW:

1.“**Fake Product Review Monitoring System**” by Piyush Jain, Karan Chheda, Mihir Jain, Prachiti Lade[1]- This technique includes Ontology, Geo location and IP address tracking, Spam words Dictionary using Naïve Bayes, Brand only review detection and tracking account used in order to detect whether the review of the product is computer generated or written by authentic user.

2.”**Opinion Spam Detection in Online Reviews**” by Ajay Rastogi and Monica Mehrotra[2]- This paper discusses the reviews based on textual and linguistic, behavioural, and relational features and further tests and discusses various state-of-the-art models used previously on the research of this topic.

3.”**Modeling Review Spam Using Temporal Patterns and Co-bursting Behaviors**” by Huayi Li , Geli Fei , Shuai Wang , Bing Liu , Weixiang Shao , Arjun Mukherjee , Jidong Shao[3]-This paper discusses a two-mode Labeled Hidden Markov Model to detect spammers and also build a co-bursting spammer detection network that detects spammers who write several reviews in a short time and of a variety of products.

AIM:

1.To develop a machine learning model to detect fake reviews written on amazon
2.Improve the credibility of the customer reviews by identifying and filtering out the fake or biased reviews
3.Enhance the overall shopping experience for consumers by promoting authentic and reliable reviews of products.

METHODOLOGY: In this section , we describe the procedure of acquiring the literature (i.e, dataset and research papers) as described in the fig. We first sieve throught the google and select only papers published by renowned authors or those papers that have or present a unique way of approaching the problem.

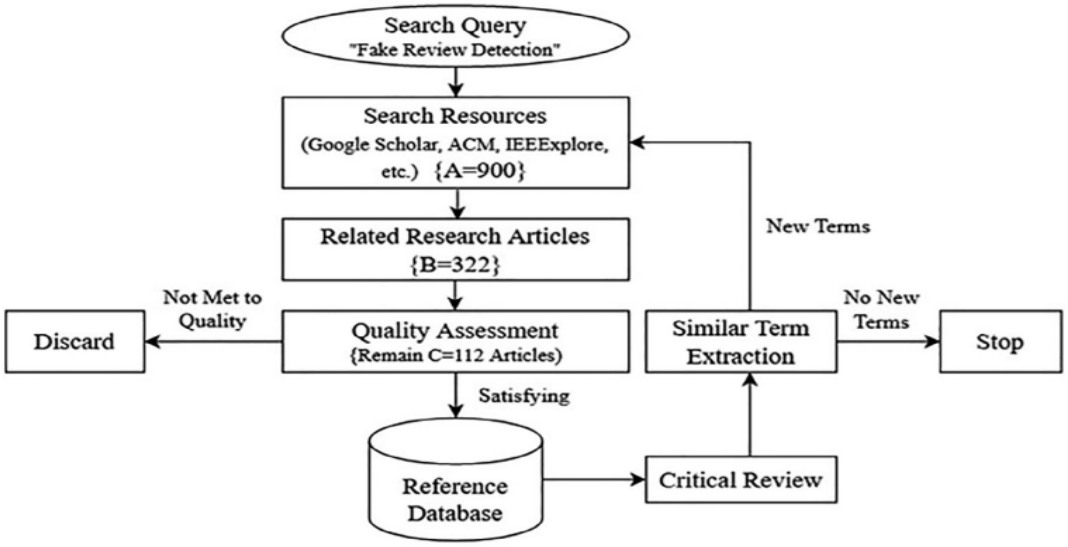


Fig. The process of Data collection

FRD(Fake Review Detection) is a sub-domain of opinion spam detection. Opinion spam detection also includes spreading rumors/hoaxes for indirect financial gain, spreading propaganda news for political gain, etc., where FRD deals with deception on online review platforms, i.e., false information in reviews posted online. It includes both spammer detection and spam (or spamming) detection problem types.

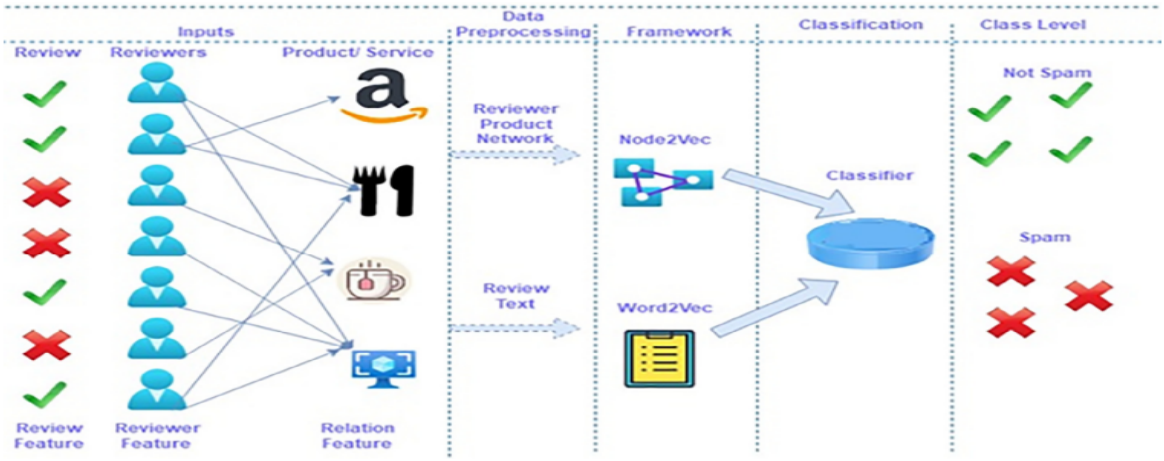


Fig. The general procedure of review spam detection

For data collection, we used the dataset containing 20k fake reviews and 20k real product reviews. OR = Original reviews (presumably human created and authentic); CG = Computer-generated fake reviews. Moving on to the data preprocessing, we used a variety of techniques.The **libraries and packages** that we used for making this project are **Numpy, Pandas, Matplotlib.pyplot, Seaborn, Warnings, nltk, nltk.corpus, String, sklearn.naive_bayes, sklearn.feature_extraction, sklearn.model_selection, sklearn.ensemble, sklearn.tree, sklearn.linear_model, sklearn.svc, sklearn.neighbors.** Next we performed the data cleaning and data preprocessing to further feed it to standard Natural Language Processing techniques then followed by a classifier model trained on to the cleaned and processed dataset to predict if the review is fake or real using various machine learning models.The techniques used for **preprocessing and cleaning** are:

1.**Removing punctuation character**

2.Transforming text to lower case

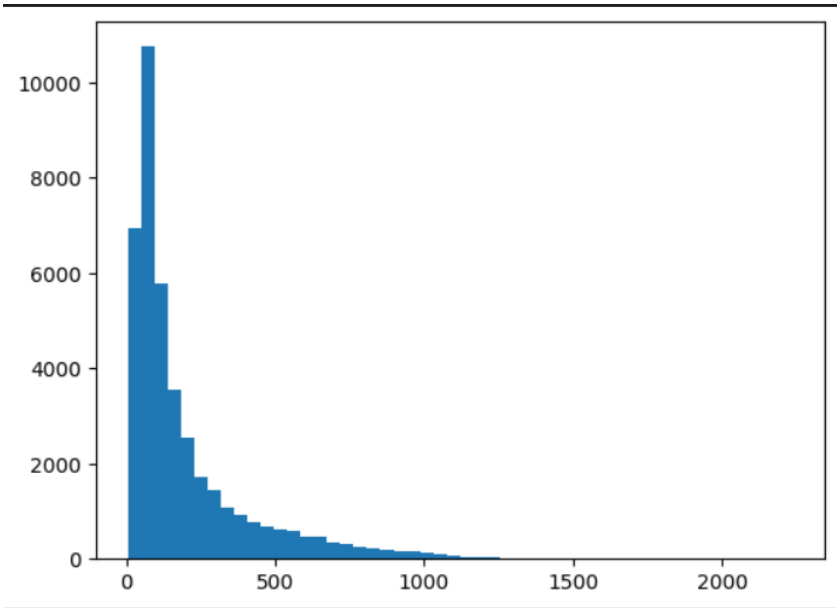
3.Eliminating stopwords

4.Stemming :the process of reducing a word to its word stem that affixes to suffixes and prefixes or the roots. While a stemming algorithm is a linguistic normalization process in which the variant forms of a word are reduced to a standard form.

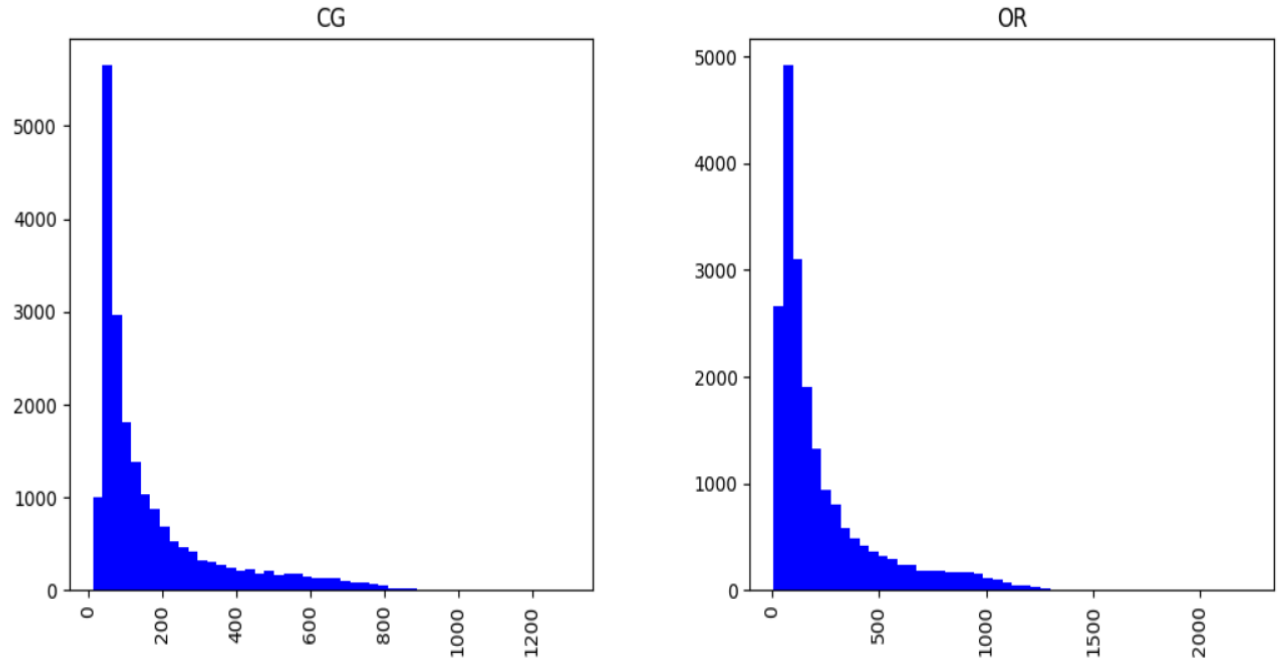
5.**Lemmatizing** : a text pre-processing technique used in natural language processing (NLP) models to break a word down to its root meaning to identify similarities.

6.**Removing digits**

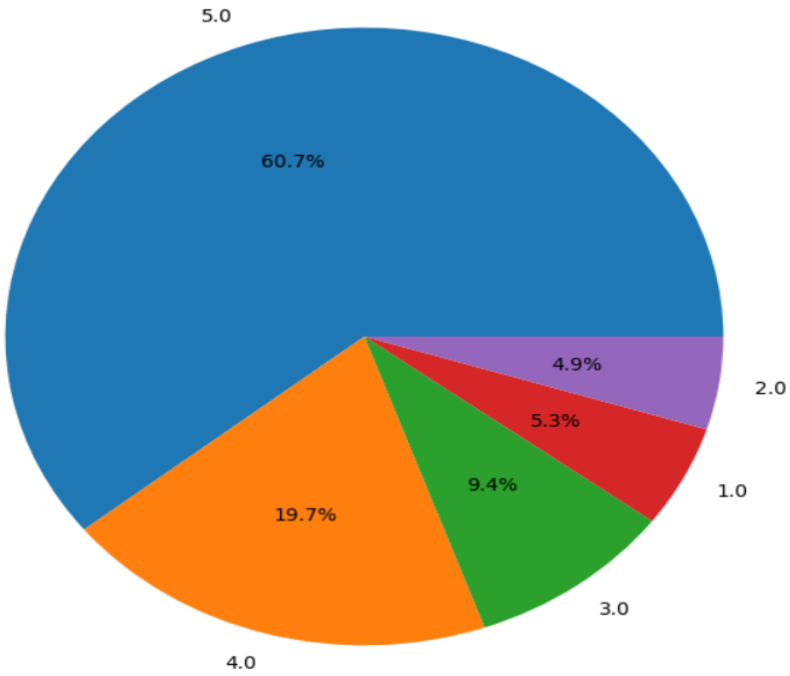
Exploratory Data Analysis:



Graph 1:Analysis of length of reviews in the dataset



Graph 2: Comparing how the length of the review differs in original review and computer generated review.



Graph 3: Pie chart representation of rating distribution among reviews

Transformers Used for Text Vectorization, Weighting and Normalization

1.**CountVectorizer Bag of Words Transformer :**Bag of words is a Natural Language Processing technique of text modelling. In technical terms, we can say that it is a method of feature extraction with text data. This approach is a simple and flexible way of extracting features from documents.

2.**TFIDF(Term Frequency-Inverse Document Frequency) Transformer :**is a widely used statistical method in natural language processing and information retrieval. It measures how important a term is within a document relative to a collection of documents

Machine Learning Algorithms Used

1.Logistic Regression :Logistic Regression is the classification technique follows under the unsupervised learning. Logistic regression is the binary.classification model which predicts the result into two values like true or false OR 1 or 0. It is used for predicting the categorical dependent variable using a given set of independent variables. When the system which results into two categories is developed the logistic regression is the best approach. It is mainly used for classification problems. The logistic regression contains two variables one is dependent variable and can call as x and another is dependent variable can call as y. x is input variable to algorithm and y is the output.

In mathematical way, y=f(x)

2.K Nearest Neighbors :The K-Nearest Neighbor (KNN) algorithm is a popular machine learning technique used for classification and regression tasks. It relies on the idea that similar data points tend to have similar labels or values.During the training phase, the KNN algorithm stores

the entire training dataset as a reference. When making predictions, it calculates the distance between the input data point and all the training examples, using a chosen distance metric such as Euclidean distance.
Next, the algorithm identifies the K nearest neighbors to the input data point based on their distances., the algorithm assigns the most common class label among the K neighbors as the predicted label for the input data point. For regression, it calculates the average or weighted average of the target values of the K neighbors to predict the value for the input data point.
The KNN algorithm is straightforward and easy to understand, making it a popular choice in various domains. However, its performance can be affected by the choice of K and the distance metric, so careful parameter tuning is necessary for optimal results.
3.Support Vector Classifier :It is a supervised machine learning algorithm that can be used for both classification or regression use cases. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is a number of features you have) with the value of each feature being the value of a particular coordinate. We perform classification by finding the hyper-plane that differentiates the two classes very well.In simpler terms, support vectors are simply the coordinates of individual observation and the SVM classifier is a frontier that best segregates the two classes (hyper-plane/line).
4.Decision Tree Classifier :A decision tree is a non-parametric supervised learning algorithm for classification and regression tasks. It has a hierarchical tree structure consisting of a root node, branches, internal nodes, and leaf nodes. Decision trees are used for classification and regression tasks, providing easy-to-understand models.
A decision tree is a hierarchical model used in decision support that depicts decisions and their potential outcomes, incorporating chance events, resource expenses, and utility. This algorithmic model utilizes conditional control statements and is non-parametric, supervised learning, useful for both classification and regression tasks. The tree structure is comprised of a root node, branches, internal nodes, and leaf nodes, forming a hierarchical, tree-like structure.
5.Random Forests Classifier : This is user-friendly nature and adaptability, enabling it to tackle both classification and regression problems effectively. The algorithm’s strength lies in its ability to handle complex datasets and mitigate overfitting, making it a valuable tool for various predictive tasks in machine learning.One of the main advantages of random forests for NLP is their ability to handle high-dimensional and sparse data sets, which are common in text analysis. Random forests can also deal with missing values, outliers, and imbalanced classes, which can affect the performance of other algorithms. Moreover, random forests are easy to implement and tune, as they have few hyperparameters and do not require much preprocessing or scaling of the data.
6.Multinomial Naive Bayes : Naive Bayes Algorithm is a family of probabilistic algorithms based on applying Bayes’ theorem with the “naive” assumption of conditional independence between every pair of a feature. Bayes theorem calculates probability P(c|x) where c is the class of the possible outcomes and x is the given instance which has to be classified, representing certain features.
P(c|x) = P(x|c) * P(c) / P(x)

Naive Bayes are mostly used in natural language processing (NLP) problems. Naive Bayes predict the tag of a text. They calculate the probability of each tag for a given text and then output the tag with the highest one.

EXPERIMENTAL RESULT:

We here have tried and tested various models with the dataset to check the accuracy of each one of them, the screenshot of the performance of each model is also attached along with the overall accuracy of the models.

```
Performance of various ML models:


Logistic Regression Prediction Accuracy: 86.4%
K Nearest Neighbors Prediction Accuracy: 57.54%
Decision Tree Classifier Prediction Accuracy: 73.51%
Random Forests Classifier Prediction Accuracy: 84.26%
Support Vector Machines Prediction Accuracy: 88.05%
Multinomial Naive Bayes Prediction Accuracy: 84.76%
```

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

          CG       0.82      0.88      0.85      7093
          OR       0.87      0.81      0.84      7058

    accuracy                           0.85     14151
   macro avg       0.85      0.85      0.85     14151
weighted avg       0.85      0.85      0.85     14151

Confusion Matrix: [[6274  819]
 [1337 5721]]
Accuracy Score: 0.8476432760935623
```

Classification Report of Multinomial Naive Bayes

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

          CG       0.81      0.89      0.85      7093
          OR       0.88      0.80      0.83      7058

    accuracy                           0.84     14151
   macro avg       0.85      0.84      0.84     14151
weighted avg       0.85      0.84      0.84     14151

Confusion Matrix: [[6311  782]
 [1446 5612]]
Accuracy Score: 0.8425552964454809
Model Prediction Accuracy: 84.26%
```

Classification Report of Random forest classifier

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

          CG       0.73      0.74      0.74      7093
          OR       0.74      0.73      0.73      7058

    accuracy                           0.74     14151
   macro avg       0.74      0.74      0.74     14151
weighted avg       0.74      0.74      0.74     14151

Confusion Matrix: [[5231 1862]
 [1887 5171]]
Accuracy Score: 0.7350717263797611
Model Prediction Accuracy: 73.51%
```

Classification Report of Decision tree

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

          CG       0.54      0.97      0.70      7093
          OR       0.87      0.17      0.29      7058

    accuracy                           0.58     14151
   macro avg       0.71      0.57      0.49     14151
weighted avg       0.71      0.58      0.49     14151

Confusion Matrix: [[6909  184]
 [5824 1234]]
Accuracy Score: 0.575436364921207
Model Prediction Accuracy: 57.54%
```

Classification Report of K nearest neighbour

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

          CG       0.89      0.86      0.88      7093
          OR       0.87      0.90      0.88      7058

    accuracy                           0.88     14151
   macro avg       0.88      0.88      0.88     14151
weighted avg       0.88      0.88      0.88     14151

Confusion Matrix: [[6129  964]
 [ 727 6331]]
Accuracy Score: 0.8805031446540881
Model Prediction Accuracy: 88.05%
```

Classification Report of support vector classifier

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

          CG       0.87      0.85      0.86      7093
          OR       0.85      0.88      0.87      7058

    accuracy                           0.86     14151
   macro avg       0.86      0.86      0.86     14151
weighted avg       0.86      0.86      0.86     14151

Confusion Matrix: [[6033 1060]
 [ 864 6194]]
Accuracy Score: 0.8640378771818246
Model Prediction Accuracy: 86.4%
```

Classification Report of Logistic Regression

CONCLUSION: Support Vector Machines Classifier performed the most accurate predictions regarding the fake nature of reviews having a predictive accuracy of just over 88%, closely followed by Logistic Regression which had a prediction accuracy of a little more than 86%. Random Forests Classifier and Multinomial Naive Bayes algorithm predicted to a precision level of approximately 84%. However, the Decision Tree Classifier performed fake reviews prediction upto an accuracy of just over 73%. The worst performing algorithm was the K Nearest Neighbors algorithm which could only perform the predictions upto an accuracy level of nearly 58%.

APPENDICES:

1.https://osf.io/tyue9/ - dataset

2.https://github.com/anjali-113/fakereviewdetection2/blob/main/fakereviewdetection.ipynb -code