Journal of Cardiovascular Disease Research

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

Semi-Supervised and Supervised Learning Based Detection of Fake Online Reviews

Sapthagiri Vienala, Santhosh Kumar Vanamala

Department of Electronics and Communication Engineering

Sree Dattha Group of Institutions, Hyderabad, Telangana, India.

Abstract

Online reviews have great impact on today's business and commerce. Decision making for purchase of online products mostly depends on reviews given by the users. Hence, opportunistic individuals or groups try to manipulate product reviews for their own interests. This paper introduces some semi-supervised and supervised text mining models to detect fake online reviews as well as compares the efficiency of both techniques on dataset containing hotel reviews.

Keywords: Fake online reviews, semi supervised learning, supervised learning.

1. Introduction

In this era of the internet, customers can post their reviews or opinions on several websites. These reviews are helpful for the organizations and for future consumers, who get an idea about products or services before selecting. In recent years, it has been observed that the number of customer reviews has increased significantly. Customer reviews affect the decision of potential buyers. In other words, when customers see reviews on social media, they determine whether to buy the product or reverse their purchasing decisions. Therefore, consumer reviews offer an invaluable service for individuals. Positive reviews bring big financial gains, while negative reviews often exert a negative financial effect. Consequently, with customers becoming increasingly influential to the marketplace, there is a growing trend towards relying on customers' opinions to reshape businesses by enhancing products, services, and marketing. For example, when several customers who purchased a specific model of Acer laptop posted reviews complaining about the low display quality, the manufacturer was inspired to produce a higher-resolution version of the laptop.

Fake reviews can be created in two main ways. First, in a (a) human-generated way by paying human content creators to write authentic-appearing but not real reviews of products — in this case, the review author never saw said products but still writes about them. Second, in a (b) computer-generated way by using text-generation algorithms to automate the fake review creation. Traditionally, human-generated fake reviews have been traded like commodities in a "market of fakes" – one can simply order reviews online in a given quantity, and human writers would carry out the work. However, the technological progress in text generation – natural language processing (NLP) and machine learning (ML) to be more specific – has incentivized the automation of fake reviews, as with generative language models, fake reviews could be generated at scale and a fraction of the cost compared to human-generated fake reviews. Yuanyuan Wu et. al [1] proposes an antecedent–consequence–intervention conceptual framework to develop an initial research agenda for investigating fake reviews. Based on a review of the extant literature on this issue, they identify 20 future research questions and suggest 18 propositions. Notably, research on fake reviews is often limited by lack of high-quality datasets. To alleviate this problem, they comprehensively compile and summarize the existing fake reviews-related public datasets. They conclude by presenting the theoretical and practical implications of the current research. Liu et. al [2] proposed a method for the detection of fake reviews based on review records associated with products. They first analyse the characteristics of review data using a crawled Amazon China dataset, which shows that the patterns of

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

review records for products are similar in normal situations. In the proposed method, they first extract the review records of products to a temporal feature vector and then develop an isolation forest algorithm to detect outlier reviews by focusing on the differences between the patterns of product reviews to identify outlier reviews. They will verify the effectiveness of our method and compare it to some existing temporal outlier detection methods using the crawled Amazon China dataset. They will also study the impact caused by the parameter selection of the review records. Our work provides a new perspective of outlier review detection, and our experiment demonstrates the effectiveness of our proposed method.

1.1 Problem Definition

In recent years, online reviews have become the most important resource of customers' opinions. These reviews are used increasingly by individuals and organizations to make purchase and business decisions. Unfortunately, driven by the desire for profit or publicity, fraudsters have produced deceptive (spam) reviews. The fraudsters' activities mislead potential customers and organizations reshaping their businesses and prevent opinion-mining techniques from reaching accurate conclusions.

2. Literature survey

Mohawesh et. al [3] presented an extensive survey of the most notable works to date on machine learning-based fake review detection. Firstly, they have reviewed the feature extraction approaches used by many researchers. Then, they detailed the existing datasets with their construction methods. Then, they outlined some traditional machine learning models and neural network models applied for fake review detection with summary tables. Traditional statistical machine learning enhances text classification model performance by improving the feature extraction and classifier design. In contrast, deep learning improves performance by enhancing the presentation learning method, algorithm's structure and additional knowledge. They also provided a comparative analysis of some neural network model-based deep learning and transformers that have not been used in fake review detection. The outcomes showed that RoBERTa achieved the highest accuracy on both datasets. Further, recall, precision, and F1 score proved the efficacy of using RoBERTa in detecting fake reviews. Finally, they summarised the current gaps in this research area and the possible future direction to get robust outcomes in this domain.

Ahmed et. al [4] proposed a fake news detection model that use n-gram analysis and machine learning techniques. They investigate and compare two different features extraction techniques and six different machine classification techniques. Experimental evaluation yields the best performance using Term Frequency-Inverted Document Frequency (TF-IDF) as feature extraction technique, and Linear Support Vector Machine (LSVM) as a classifier, with an accuracy of 92%.

Atefeh Heydari et. al [5] proposed a robust review spam detection system. A detailed literature survey has shown potential of the timing element when applied to this domain and lead to the development of review spam detection approach based on time series analysis methods. Based on the consideration that the capture of burst patterns in reviewing process can improve the detection accuracy, in this experiment, they propose a review spam detection approach which investigates bogusness of reviews fallen.

Paul et. al [6] reviews the literature on Fake Review Detection (FRD) on online platforms. It covers both basic research and commercial solutions, and discusses the reasons behind the limited level of

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

success that the current approaches and regulations have had in preventing damage due to deceptive reviews.

Deng et. al [7] analysed all the characteristics of fake reviews of hype and find that the text of the review always tells us the truth. For the reason that hype review is always absolute positive or negative, they proposed an algorithm to detect online fake reviews of hype about restaurants based on sentiment analysis. In our experiment, reviews are considered in four dimensions: taste, environment, service and overall attitude. If the analysis result of the four dimensions is consistent, the review will be categorized as a hype review. Our experiment results have shown that the accuracy of our algorithm is about 74% and the method proposed in this article can also be applied to other areas, such as sentiment analysis of online opinion in emergency management of emergency cases.

Rathore et. al [8] propose a top-down framework for candidate fake reviewer groups' detection based on the DeepWalk approach on reviewers' graph data and a (modified) semi supervised clustering method, which can incorporate partial background knowledge. They validate our proposed framework on a real review dataset from the Google Play Store, which has partial ground-truth information about 2207 fraud reviewer-ids out of all 38 123 reviewer-ids in the dataset. Our experimental results demonstrate that the proposed approach is able to identify the candidate spammer groups with reasonable accuracy. The proposed approach can also be extended to detect groups of opinion spammers in social media (e.g. fake comments or fake postings) with temporal affinity, semantic characteristics, and sentiment analysis.

Khan et. al [9] proposed a supervised learning-based technique for the detection of fake reviews from the online textual content. The study employs machine learning classifiers for bifurcating fake and genuine reviews. Experimental results are evaluated against different evaluation measures and the performance of the proposed system is compared with baseline works.

Li et. al [10] propose the concept of review group and to the best of our knowledge, it's the first time the review group concept is proposed and used. Review grouping algorithm is designed to effectively split reviews of reviewer into groups which participate in building novel grouping models to identify both positive and negative deceptive reviews. Several new features which are language independent based on group model are constructed. Additionally, they explore the collusion relationship between reviewers to build reviewer group collusion model. Evaluations show that the review group method and reviewer group collusion models can effectively improve the precision by 4%–7% compared to the baselines in fake reviews classification task especially when reviews are posted by professional review spammers.

3. Proposed system

In this project, we make some classification approaches for detecting fake online reviews, some of which are semi-supervised, and others are supervised. For semi-supervised learning, we use Expectation-maximization algorithm. Statistical Naive Bayes classifier and Support Vector Machines (SVM) are used as classifiers in our research work to improve the performance of classification. We have mainly focused on the content of the review-based approaches. As feature we have used word frequency count, sentiment polarity and length of review.

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

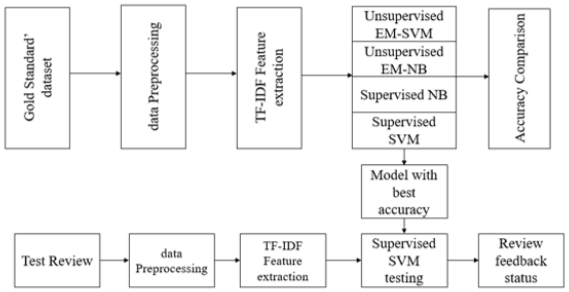


Fig. 1: Block diagram of proposed system.

Dataset description

2- Columns: Review, Lebel

1601-Rows

Data Pre-processing in Machine learning

Data pre-processing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data pre-processing task.

Why do we need Data Pre-processing?

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data pre-processing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

- Getting the dataset
- Importing libraries
- Importing datasets
- Finding Missing Data
- Encoding Categorical Data
- Splitting dataset into training and test set
- Feature scaling

Splitting the Dataset into the Training set and Test set

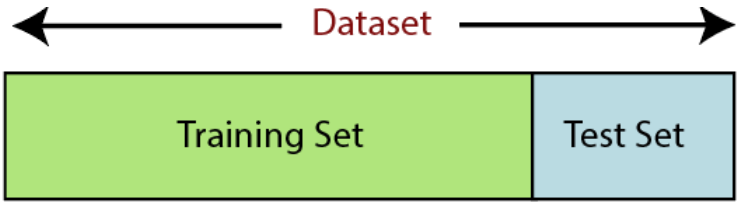
In machine learning data pre-processing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model.

Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So, we always try to make a machine learning

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:



Training Set: A subset of dataset to train the machine learning model, and we already know the output.

Test set: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

TF-IDF

TF-IDF which stands for Term Frequency – Inverse Document Frequency. It is one of the most important techniques used for information retrieval to represent how important a specific word or phrase is to a given document. Let's take an example, we have a string or Bag of Words (BOW) and we have to extract information from it, then we can use this approach.

The tf-idf value increases in proportion to the number of times a word appears in the document but is often offset by the frequency of the word in the corpus, which helps to adjust with respect to the fact that some words appear more frequently in general. TF-IDF use two statistical methods, first is Term Frequency and the other is Inverse Document Frequency. Term frequency refers to the total number of times a given term t appears in the document doc against (per) the total number of all words in the document and the inverse document frequency measure of how much information the word provides. It measures the weight of a given word in the entire document. IDF show how common or rare a given word is across all documents. TF-IDF can be computed as tf * idf

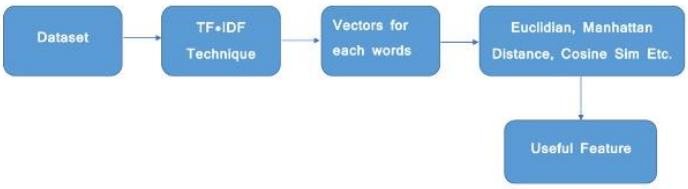


Fig. 2: TF-IDF block diagram.

TF-IDF do not convert directly raw data into useful features. Firstly, it converts raw strings or dataset into vectors and each word has its own vector. Then we'll use a particular technique for retrieving the feature like Cosine Similarity which works on vectors, etc.

Terminology

t — term (word)

d — document (set of words)

N — count of corpus

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

corpus — the total document set

Step 1: Term Frequency (TF): Suppose we have a set of English text documents and wish to rank which document is most relevant to the query, "Data Science is awesome!" A simple way to start out is by eliminating documents that do not contain all three words "Data" is", "Science", and "awesome", but this still leaves many documents. To further distinguish them, we might count the number of times each term occurs in each document; the number of times a term occurs in a document is called its term frequency. The weight of a term that occurs in a document is simply proportional to the term frequency.

$$tf(t,d) = count\ of\ t\ in\ d\ /\ number\ of\ words\ in\ d$$

Step 2: Document Frequency: This measures the importance of document in whole set of corpora, this is very similar to TF. The only difference is that TF is frequency counter for a term t in document d, whereas DF is the count of occurrences of term t in the document set N. In other words, DF is the number of documents in which the word is present. We consider one occurrence if the term consists in the document at least once, we do not need to know the number of times the term is present.

$$df(t) = occurrence\ of\ t\ in\ documents$$

Step 3: Inverse Document Frequency (IDF): While computing TF, all terms are considered equally important. However, it is known that certain terms, such as "is", "of", and "that", may appear a lot of times but have little importance. Thus, we need to weigh down the frequent terms while scale up the rare ones, by computing IDF, an inverse document frequency factor is incorporated which diminishes the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely. The IDF is the inverse of the document frequency which measures the informativeness of term t. When we calculate IDF, it will be very low for the most occurring words such as stop words (because stop words such as "is" is present in almost all of the documents, and N/df will give a very low value to that word). This finally gives what we want, a relative weightage.

$$idf(t) = N/df$$

Now there are few other problems with the IDF, in case of a large corpus, say 100,000,000, the IDF value explodes, to avoid the effect we take the log of idf . During the query time, when a word which is not in vocab occurs, the df will be 0. As we cannot divide by 0, we smoothen the value by adding 1 to the denominator.

$$idf(t) = log(N/(df + 1))$$

The TF-IDF now is at the right measure to evaluate how important a word is to a document in a collection or corpus. Here are many different variations of TF-IDF but for now let us concentrate on this basic version.

$$tf - idf(t,d) = tf(t,d) * log(N/(df + 1))$$

Step 4: Implementing TF-IDF: To make TF-IDF from scratch in python, let's imagine those two sentences from different document:

first sentence: "Data Science is the sexiest job of the 21st century".

second sentence: "machine learning is the key for data science".

Expected Maximization (EM)

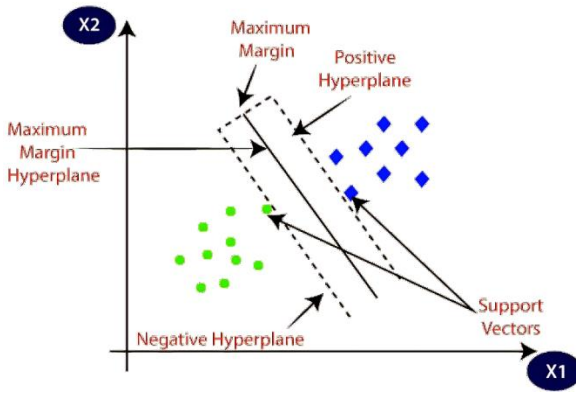
ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

The Expectation-Maximization (EM) algorithm is a way to find maximum-likelihood estimates for model parameters when your data is incomplete, has missing data points, or has unobserved (hidden) latent variables. It is an iterative way to approximate the maximum likelihood function.

Support Vector Machine Algorithm (SVM)

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



SVM Algorithm: Machine learning involves predicting and classifying data and to do so we employ various machine learning algorithms according to the dataset. SVM or Support Vector Machine is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems. The idea of SVM is simple: The algorithm creates a line or a hyperplane which separates the data into classes. In machine learning, the radial basis function kernel, or RBF kernel, is a popular kernel function used in various kernelized learning algorithms. In particular, it is commonly used in support vector machine classification. As a simple example, for a classification task with only two features (like the image above), you can think of a hyperplane as a line that linearly separates and classifies a set of data.

- Intuitively, the further from the hyperplane our data points lie, the more confident we are that they have been correctly classified. We therefore want our data points to be as far away from the hyperplane as possible, while still being on the correct side of it.
- So, when new testing data is added, whatever side of the hyperplane it lands will decide the class that we assign to it.

Applications

- Face recognition
- Weather prediction
- Medical diagnosis

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

- Spam detection
- Age/gender identification
- Language identification
- Sentimental analysis
- Authorship identification
- News classification

Naïve Bayes (NB)

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems. ... Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

To implement this project, we have used 'Gold Standard' Dataset which contains 1600 reviews from which 800 are genuine reviews and 800 are fake reviews and to train both supervised and semi supervised we have used this dataset and this dataset saved inside 'Dataset' folder and below screen shots showing dataset details.

Advantages of proposed system

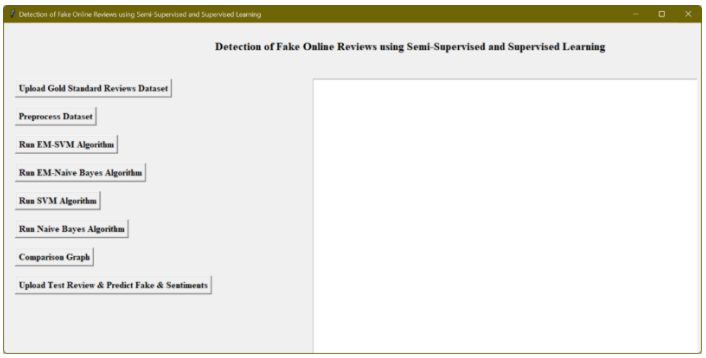
- The system is very fast and effective due to semi-supervised and supervised learning.
- Focused on the content of the review-based approaches. As feature we have used word frequency count, sentiment polarity and length of review.

4. Results

Module Implementation

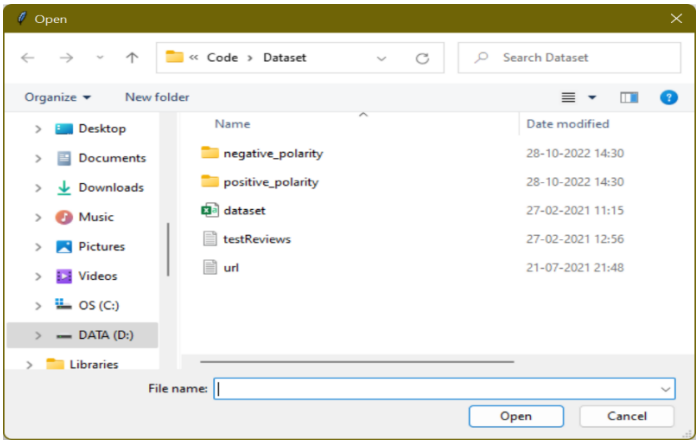
To implement this project, we have designed following modules

- Upload Reviews Dataset
- Pre-process Dataset
- Run EM-SVM Algorithm
- Run EM-Naive Bayes Algorithm
- Run SVM Algorithm
- Run Naive Bayes Algorithm
- Comparison Graph
- Upload Test Review & Predict Fake & Sentiments

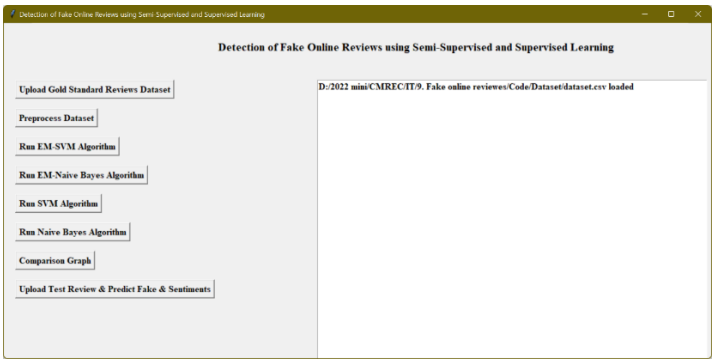


ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

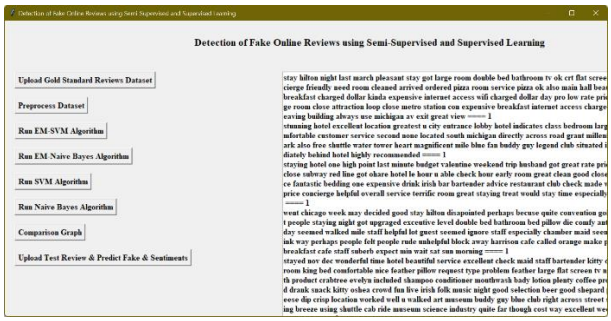
In above screen click on 'Upload Gold Standard Reviews Dataset' button to upload dataset



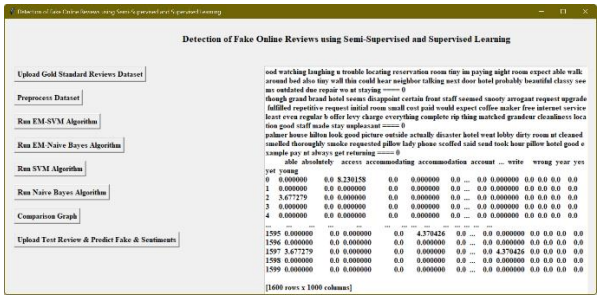
In above screen selecting and uploading 'dataset.csv' file and then click on 'Open' button to load dataset and to get below screen



In above screen dataset loaded and now click on 'Preprocess Dataset' button to read and process dataset and to get below screen

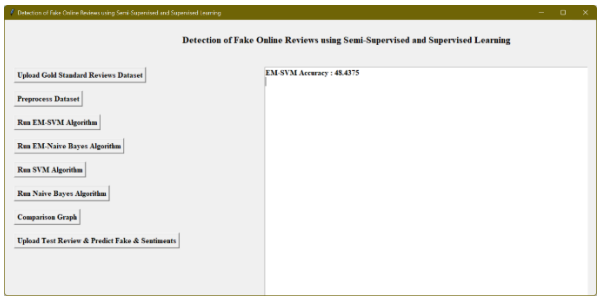


In above screen from all reviews, we removed stop words and after === symbol we can see it label as 0 or 1 and now scroll down above screen to bottom to see TF-IDF vector. You can see below screen

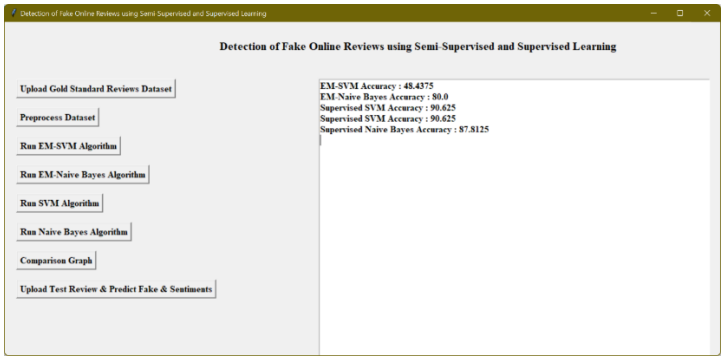


ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

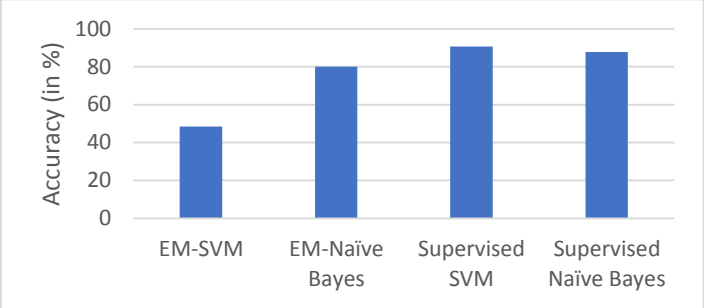
In above screen in top row, we can see all words separated by TAB and in below top row we can see its numeric value calculated using TF-IDF. In above screen in bottom, we can see dataset contains total 1600 reviews and then application using 1280 reviews for training and 320 reviews for testing. Now train and test data is ready and now click on 'Run EM-SVM Algorithm' button to train it



In above screen EM-SVM got 48.43% accuracy and similarly click next 3 buttons to train all algorithms

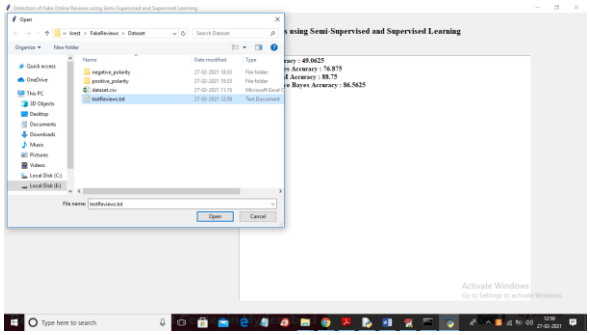


In above screen we can see EM algorithms are not working well but supervise algorithms are giving better accuracy and now click on 'Comparison Graph' button to get below graph

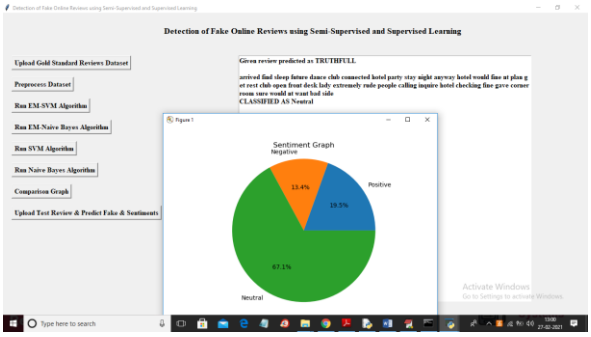


In above screen x-axis represents algorithm name and y-axis represents accuracy of those algorithms and from above graph we can say supervised algorithms are better than EM. Now click on 'Upload Test Review & Predict Fake & Sentiments' button to upload test review and to get below output for each review

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020



In above screen selecting and uploading 'testReviews' and then click on 'Open' button to get below result



In above screen we can see review detected as TRUTHFULL and its sentiment predicted as NEUTRAL

5. Conclusion

We have shown several semi-supervised and supervised text mining techniques for detecting fake online reviews in this research. We have combined features from several research works to create a better feature set. Also, we have tried some other classifier that were not used on the previous work. Thus, we have been able to increase the accuracy of previous semi supervised techniques done by Jiten et al. We have also found out that supervised Naive Bayes classifier gives the highest accuracy. This ensures that our dataset is labeled well as we know semi-supervised model works well when reliable labeling is not available. In our research work we have worked on just user reviews.

5.1 Future scope

In future, user behaviors can be combined with texts to construct a better model for classification. Advanced preprocessing tools for tokenization can be used to make the dataset more precise. Evaluation of the effectiveness of the proposed methodology can be done for a larger data set.

References

[1] Yuanyuan Wu, Eric W.T. Ngai, Pengkun Wu, Chong Wu, Fake online reviews: Literature review, synthesis, and directions for future research, Decision Support Systems, Volume 132, 2020, 113280, ISSN 0167-9236, https://doi.org/10.1016/j.dss.2020.113280.

[2] W. Liu, J. He, S. Han, F. Cai, Z. Yang and N. Zhu, "A Method for the Detection of Fake Reviews Based on Temporal Features of Reviews and Comments," in IEEE Engineering Management Review, vol. 47, no. 4, pp. 67-79, 1 Fourthquarter,Dec. 2019, doi: 10.1109/EMR.2019.2928964.

ISSN: 0975-3583, 0976-2833 VOL11, ISSUE04, 2020

[3] R. Mohawesh. "Fake Reviews Detection: A Survey," in IEEE Access, vol. 9, pp. 65771-65802, 2021, doi: 10.1109/ACCESS.2021.3075573.

[4] Ahmed, H., Traore, I., Saad, S. (2017). Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. In: Traore, I., Woungang, I., Awad, A. (eds) Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments. ISDDC 2017. Lecture Notes in Computer Science (), vol 10618. Springer, Cham. https://doi.org/10.1007/978-3-319-69155-8_9

[5] Atefeh Heydari, Mohammadali Tavakoli, Naomie Salim, Detection of fake opinions using time series, Expert Systems with Applications, Volume 58, 2016, Pages 83-92, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2016.03.020.

[6] Paul, H., Nikolaev, A. Fake review detection on online E-commerce platforms: a systematic literature review. Data Min Knowl Disc 35, 1830–1881 (2021). https://doi.org/10.1007/s10618-021-00772-6

[7] Deng, X., Chen, R. (2014). Sentiment Analysis Based Online Restaurants Fake Reviews Hype Detection. In: Han, W., Huang, Z., Hu, C., Zhang, H., Guo, L. (eds) Web Technologies and Applications. APWeb 2014. Lecture Notes in Computer Science, vol 8710. Springer, Cham. https://doi.org/10.1007/978-3-319-11119-3_1

[8] P. Rathore, J. Soni, N. Prabakar, M. Palaniswami and P. Santi, "Identifying Groups of Fake Reviewers Using a Semisupervised Approach," in IEEE Transactions on Computational Social Systems, vol. 8, no. 6, pp. 1369-1378, Dec. 2021, doi: 10.1109/TCSS.2021.3085406.

[9] Khan, H., Asghar, M.U., Asghar, M.Z., Srivastava, G., Maddikunta, P.K.R., Gadekallu, T.R. (2021). Fake Review Classification Using Supervised Machine Learning. In: , et al. Pattern Recognition. ICPR International Workshops and Challenges. ICPR 2021. Lecture Notes in Computer Science (), vol 12664. Springer, Cham. https://doi.org/10.1007/978-3-030-68799-1_19

[10] Li, Y., Wang, F., Zhang, S. et al. Detection of Fake Reviews Using Group Model. Mobile Netw Appl 26, 91–103 (2021). https://doi.org/10.1007/s11036-020-01688-z