****

BERZIET UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

**Review Spam Detection Based On Hybrid Features**

Prepared by

Muhannad Shamasneh

Mohammad Hussein

Mohammad Abu Mahmoud

Supervised by

Hanna Bullata, Aziz Qarush.

A graduation Project submitted to the Department of Electrical

and Computer Engineering in partial fulfillment of the requirements

for the degree of B.Sc. in Electrical Engineering

BIRZEIT

April – 2018

الإهداء

**المستخلص**

**ازدادت بشكل ملحوظ مؤخراً المواقع التجارية التي تسمح للمستخدمين شراء السلع او الخدمات عن طريق الانترنت. هذه المواقع تحتوي على معلومات ذات أهمية كبيرة للمشترين و أصحاب السلع أو الأعمال، ألا وهي الآراء التي يكتبها المستخدمون على هذه السلع أو الخدمات بحيث تعكس تجربتهم مع هذه السلعة ورأيهم فيها من حيث ايجابياتها أو سلبياتها. هذه الآراء تعد مهمة جداً للمستخدمين وأصحاب الأعمال، فمن ناحية المستخدم فانه قبل شراء اي منتج فإنه غالبا ما يلجأ الى قراءة آراء الناس حول هذا المنتج وان هذه الأراء تؤثر على قرار الشراء عنده. أما بالنسبة لأصحاب الأعمال فان حجم مبيعاتهم يتأثر زيادةً او نقصاناً من خلال هذه الآراء ان كانت ايجابية أو سلبية. ولذلك فقد لجأ بعض المستخدمين أو أصحاب الأعمال الى كتابة أراء مزيفة لترويج لمنتجاتهم أو لنقد منتجات المنافسين وذلك للتأثير على القرار الشرائي للمستخدم.**

**في هذا التقرير فاننا نعرض لكم طريقة كشف الآراء المزيفة والمستخدمين الذين يكتبون آراء مزيفة باستخدام هجين من المميزات التي تميز هذه الآراء الزائفة عن غيرها من الآراء، وايضاً كشف الذين يقومون بكتابة آراء مزيفة بحيث اننا قمنا بتجهيز البيانات المتوفرة لدينا ومن ثم استخلصنا منها عدة ميزات تميز النص المزيف والأشخاص المزيفين. ثم استخدمنا ثلاثة طرق لتصنيف البيانات: الأولى ، وهي الطريقة التقليدية لتصنيف البيانات باستخدام التعليم الآلي. الطريقة الثانية تعتمد على نموذج احتمالي الذي يعتمد على خليط بين مميزات الأشخاص المزيفين والنص المزيف، والطريقة الثالثة تعتمد على نظام النتيجة بحيث يتم اعطاء نتيجة للنص أو الشخص بناءً على الميزات له.**

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Abstract**

Commercial websites that allow people to purchase products and services online through internet recently rapidly grown. These websites contain a lot of user reviews regarding the products and services they used. These reviews contain valuable information about the products itself and customer experience. Therefore, this source of information became very critical for the customer and business. For the customer, his purchase decision is highly affected by the reviews from other customers since they tend to read the reviews for the product before taking any purchase decision and compare it with other products. For business, their sales can be affected by the reviews since positive reviews for the product attracts more customers also, negative reviews will affect the number of customers who intend to buy the product. Due to this importance some users or businesses illegally tends to write fake reviews to promote their products or demote competitors’ products so this problem has been taken into consideration of researches nowadays.

In this Report we introduce a methodology based on hybrid features for detecting spam reviews and spammers. It focused on extracting a set of informative features for both review text and reviewer account after doing preprocessing for the dataset. Then we define our models for detecting the spam reviews or spammers including classical classification using supervised machine learning, probabilistic model optimized by genetics and score-based model optimized by genetics.

Table of Contents

**Dedication1**

**Abstract in English1**

**Abstract in Arabic1**

**1 Introduction1**

1.1 Introduction and motivation2

1.2 Problem Statement3

1.3 Methodology2

1.4 Report outline2

**Chapter 2 Related Work4**

2.1 Review Spam Detection5

2.1.1 Fake review6

2.1.2 Non-Related6

2.1.3 Review on Brand6

2.2 Spammer Detection5

2.2.1 Based on Review Text6

2.2.2 Based on Reviewer Account Meta Data6

2.2.2 Based on Graph features6

**Chapter 3 Proposed Work4**

3.1 Notation5

3.2 Feature Extraction5

3.2.1 Text Features6

3.2.2 Spammer Features6

3.3 Review Classification5

3.3.1 Classical Machine Learning6

3.3.2 Probabilistic Model Optimized by Genetics Algorithm6

3.3.3 Score Based Model Optimized by Genetics Algorithm6

**Chapter 4 Datasets, Tools and Evaluation Measures 4**

3.1 Dataset 5

3.2 Tools 5

3.3 Evaluation Measures 5

**Chapter 5 Conclusion****4**

**References4**

**List of Figures**

[Figure 1.1.1: Type of Spam 1](#_Toc514347065)

[Figure 1.1.2: Example of Fake Review Type 2](#_Toc514347066)

[Figure 2.1.1: Spam detection techniques based on used feature 5](#_Toc514347068)

[Figure 2.2.1.1.1: Review preprocessing stages. 19](#_Toc514347069)

[Figure 3 20](#_Toc514347070)

[Figure 3.3.3.1.1: Genetics algorithms steps 26](#_Toc514347071)

[Figure 4.3.1.1: Confusion Matrix 31](#_Toc514347072)

[Figure 4.3.2.1: ROC curve evaluation measure. 33](#_Toc514347073)

**List of Equations**

**Equations 3.2.1.1.1: Vector Space Model Representation………………………………….**

**Equations 3.2.1.1.2: Cosine Similarity………………………………………………….**

**Equations 3.2.1.3: Length of the Review Feature…………………………………………….**

**Equations 3.2.2.1: Normalized Maximum Number of Reviews Feature………………….**

**Equations 3.2.2.3: Helpfulness rating for the Review……………………………………….**

**Equations 3.2.2.3: Account Freshness Feature………………………………………………….**

**Equations 3.2.2.2: Reviewing Early Feature………………………………………………….**

**Equations 3.2.2.4 Rating Deviation Score Feature………………………………………….**

**Equations 3.3.2 Probabilistic model………………………………………………….**

**Equations 3.3.3 Score based model ………………………………………………….**

**Equations 4.3.2 Accuracy ………………………………………………….**

**Equations 4.3.3 Precision………………………………………………….**

**Equations 4.3.4 Recall ………………………………………………….**

**Equations 4.3.5 F-measure ………………………………………………….**

**Equations 4.3.6 ROC curve ………………………………………………….**

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Chapter I**

**Introduction**

**Introduction and motivation**

The Internet is now considered to be the most demanded service all around the world. People now can communicate and share their knowledge very easily through different ways such as writing comments in the social sites or review the products they have used describing their experience using that product. As a result, the reviews became very important for the customer and the business --as many online shopping paradigms have spread out which help users to review products and share their reviews on social media. Such reviews impose a huge impact on product sales since people tend to check the reviews before deciding to purchase products. Since reviews can be provided by anyone on the public domain, users misuse the reviewing services by providing fake reviews to promote or demote particular products. Later in this research, fake reviews will be referred as ‘spam’ and individuals who provide spam reviews will be referred as spammers.

|  |  |  |
| --- | --- | --- |
|  |  |  |

Figure .1: Type of Spam

Spams are mainly classified as web spam, Email spam and spam reviews as shown in the figure 1.1. Web spam is defined as any web page content that used to improve the ranking of this page more than the others. Web spam is also classified as content spam which means adding some phrases to the web page document to increase its rank and link spam by posting the link of site that the spammer wants to promote or demote. While the E-mail spam is defined as unwanted or unrelated which may be used for advertisement or for different purposes. However, the E-mail spam can be detected easily because it has a special component so it has less detriment than web spam. Finally, the spam reviews (i.e. fake reviews provided by spammers to promote or demote particular product) is the type to be covered in this research. Spam reviews are classified by Jindal et al [1] into mainly three types:

1. **Type 1 Untruthful reviews:** Providing fake opinions about products to promote or demote products by undeserving positive and negative opinion respectively which is written with hidden motives. This type of reviews is also called fake reviews, Figure 1.2 shows an example on type 1.

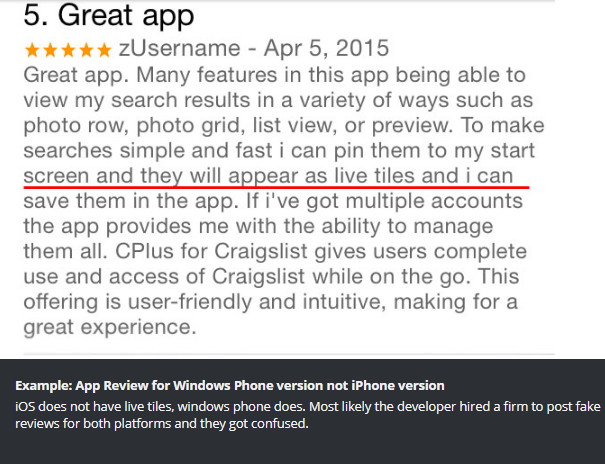


Figure 1.1.2: Example of Fake Review Type

1. **Type 2 Review on brands:** this type of spam not sufficient to concentrate on the products or services, since it can be on the cost of product, the sellers, the manufacturers of the products and other aspects. Although it supposed to be on true, but they are considered as spam as they cannot provide any helpful about the Concerned product. (e.g. “I hate Dell product and I’ll never buy their products”).
2. **Type 3 Non-Reviews:** it also called Non-related reviews which are considered as spam since it can be an advertisement or unhelpful opinions such as answers, question and any random text. (e.g. “<3 <3 <3 <3 <3” and “how to make a deliver request?”).

Due to the big challenge in the business today some companies illegally tend to hires employees to write fake reviews to promote their products or demote other competitor product. So, there is a big concern from the customer and the business to have real reviews, for these reasons many sites tend to filter spams and display only the real reviews in order to maintain credibility. Handling the problem of review spam can be done in two ways manually filtering which is time consuming and will not be effective enough because the spammers now use new methods to write spam reviews that’s hard to detect by human, the second way is automated filtering based on a set of features extracted from the review text and the reviewer accountto identify the spam reviews. One of the examples from websites that uses automated filtering is YELP[[1]](#footnote-1) since they use filters to detect the spam reviews made on the website and just showing to the user’s honest reviews.

**1.2 Problem Statement**

Spam reviews problem is now considered to be challenge problem Since most people nowadays use online shopping and most of their purchase decisions are affected by the reviews on that product, especially the first reviews appears in the websites, therefore, the customer looking for honest reviews to take an informative decision.

Our goal is to develop an automated solution to filter the reviews based on extracted features from the review content, reviewer behaviors, and review/ reviewer information. This application will yield the customers to trust the websites that uses spam filters and just display the honest reviews because they can now take an informative decision to buy or not from a set of non-spam reviews.

**1.3 Methodology**

There are four major steps to create our spam detection model which is based on a set of features extracted from review text and reviewer account.

(1) Reviews gathering and annotation.

(2) Preprocessing the reviews text.

(3) Extracting hybrid features from both review text and reviewer account and representing them into a features vector.

(4) Classifying reviews into spam or not using three approaches.

a. Classical supervised machine learning techniques.

b. Probabilistic model optimized by genetics.

c. Score based model optimized by genetics.

**1.4 Report Outline**

In Chapter 2 will introduce the related work of other people in this field and the different approaches they use to solve the problem. Chapter 3 discusses our proposed work, features we hire for detecting spam reviews, and the classification models for the reviews. Chapter 4 discusses the different choices for the data set and advantages/disadvantages for them. Then we introduce our choice among these datasets and why we choose it. Also, the tools will be used and why among different available tools in this field and the measures to evaluate the results. Lastly, chapter 5 a conclusion remarks for the overall results compared with related work and introduces the future work with possible improvements.

**Chapter II**

**Related work**

In this chapter we will introduce the work of other people in this field. We classify this chapter into two main sub sections, the first one is Spam review detection in which how people identify the spam depends only on the data and metadata for the review itself. The second section is about identifying spammers (reviewers who involved in spamming activities) as in the figure () below, also, to explain the datasets used and how to annotate the datasets.

Figure 2.1: Spam detection techniques based on used feature

**2.1 Spam Review Detection**

These studies are about detecting spam reviews using the features extracted from the review text and its metadata. We divide the work here into 3 subsections which they are identifying the fake review(type1), review on brands (type 2), non-related reviews (type 3), and hybrid from them.

**2.1.1 Fake review**

**Jindal et al.** (1)focused on identifying the unexpected patterns of reviewers which in turn used to detect the spam reviews, so this paper divides the data set into two classes spam or not spam by states if a reviewer wrote only a highly negative reviews on a specific product while many others wrote in general some positive reviews so this review is highly expected to be spam. Hence, this paper is to finding an application that based on a heuristic program to find the unexpected behavior, so this approach firstly defining several types of expectation based on the data itself by giving joint probabilities to determine the unexpectedness (deviation from expectations), then to give the unexpectedness a measure to rank the review. It then based on different measurement identify if this review is spam or not.

**Ott et al.** (2)adopt lexical model and used Bag of words which is the most common features used in this model. This consists of group of words that is extracted from the reviews text from which n-gram features can be extracted, the researchers developed a model relying on the part of speech, Bigram and Linguistic Inquiry and Word Count features, they also used the frequency of n-gram in their model to detect the un-truth review (type 1). They used a golden standard dataset from amazon Mechanical Turk about hotels, and also a data of honest reviews from Tip Adviser, then they used unsupervised classifier using SVM classifier to divide all the opinions (deceptive and honest reviews) into positive and negative, this step is used to separate the negative and positive data to achieve 86% accuracy and when they remove the separation they achieved 84% accuracy this implies that separation of data is improve the accuracy from 84% to 86%.

**Qian & Lie et al.** (3)Used conceptual level Similarity Measure based Review Spam Detection in this approach of spam detection the customer reviews is collected then it passed through a preprocessing stage then to a stop word removal stage and it is then try to find the duplicate of near duplicate of reviews and this is done by finding a matched features between the reviews and give a high score as the review is more matching other reviews. Then according to this score, we determine if the review is spam or not since of this score of matched features is less than a specific threshold its classified as not spam. There is a second case of not spam if the review is unique. and these two types of not spam classification is done by three steps: the first step is to extract features from reviews and storing them in features database, then the matched features that extracted previously used to construct a feature matrix, then to matching the feature calculation between reviews and that is done by calculating a score for different review pairs, and then to categorized as spam or non-spam based on threshold value.

**Ahmed H et al** (4)proposed an approach to detect opinion spam based on text analysis using semantic similarity metrics and n-gram features since it effective at detecting content similarity, then to validate the effectiveness of the n-gram features at detecting spams. They used the unique reviews that written by different users (singular reviews), they preprocessing the data, remove the stop words and transform the tokens into standard

form(stemming). Four types of n-gram features was extracted (bigrams, trigram, and four-gram). Then these features were used in machine learning classification by applying many classification methods namely, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression (LR), Linear Support Vector Machine (LSVM), Decision tree (DT) and Stochastic Gradient Descent (SGD). they use a semantic similarity metrics between the text of reviews to identify the near-duplicate or duplicate reviews. They also combined the n-gram features with the benefit of keystroke dynamic features to identify the fake reviews. Such as the writing speed per character of number based on the pauses between strokes. Since purses in typing and time while the reviewers editing their reviews patterns can help to identify the truthful and deceptive writings.

**Peng & Zhong et al** (5) proposed a sentiment analysis technique for purposes of detection. Believed from the authors that the rating will not necessarily represent the sentiment accurately. The reviews dataset was extracted from Resellerrating.com, nearly 2500 reviews was used. Information gain theory was chosen for feature selection. The information gain of the sentiment words is calculated after preprocessing. 220 positive words and 150 negative word was chosen. These lexicons have been built by SentiWordNet plus. Then to determine the sentiment strength of each review by extract features of each product. i.e., Sentiment Score, SS which means the sentiment polarity of review, Sentiment Ratio, SR which is a ratio of sentiment sentence to all sentences and Difference of Sentiment Polarity, DSP which represent if there are a conflict between rating score and sentiment score. Then to find unexpected patterns in product review by set of discriminative rules. Beside establishing a time series to detect spams which depends on rating score as indicator. Finally, they combine the time series method with discriminative rules to get more accurate results.

**2.1.2 non-related reviews**

**Mevada et al.** (6)developed system to detect the three types of review spam using supervised machine learning technique which is SVM classifier. The SVM classifier builds models for the training data to predict any future input. They divide the system into components and the first component is data processor which is responsible for collecting and cleaning the data before using the system. Then the data proceed to the next stage of text mining by remove, sentence splitter, stop words, tokenization, and POS tagging. The last component is the classifier which is trained by the data collected and tested by test data. they use SVM classifier . The input for the system is review text and after text mining component the SVM classifier calculate to any model this review is closer then give the result as spam or not.

**Long et al.** (7)proposed a method to find non-reviews based on four features. Firstly, from finding unusual patterns. Unusual patterns defined as links to other sites, phones, advertisements, email address and prices. Since as unusual patterns appears more in the review, the review much like to be a spam of non-related type. Secondly, used the opinion word ratio since true spam review needs to show a useful opinion of product, so each review should have the minimum number of key words that related to the product itself, so non-reviews contains little number of related keywords. Thirdly, based on ontology word ratio which is a base knowledge about the product since the review that cannot achieve this knowledge maybe identified as non-review. Finally, non-review spam can be identified based on sentence ratio since there is number of non-reviews can be written with standard words without a special characters and unexpected patterns mentioned previously, then to make a combination of these words that is usually meaningless. Then they used a grammar parser to know if the test of review is review or not based on the sentences list that generated.

**2.1.3 review on brand**

**Sushant & Bharat** (8)This paper talks about the detection of the first and the second type of spam reviews, the system talked about in this paper uses the n-gram model to detect the fake reviews and the feature selection for the detection of the brand spam reviews, this system also calculates a spam degree for each reviewer depending on the metadata provided in the dataset. according to the authors the system lets the user search for the product they want and APIs fitch the reviews of that product from that website (amazon.com), the system then clusters the reviews in groups, after that a generated ARFF file that contains features of an original review goes to the J48 classifier, which is used for training and testing in that system. this paper actually finds the first type of review spam and by using a classifier and applies the output of the classifier after doing the preprocessing (getting red of the stop words and stemming) to a method which checks the **word support count**and compare it with predefined threshold, deciding that the review with support count greater than the threshold is a brand spam.

**Long et al.** (7)this model used the ontology that mentioned in section 2.1.2, each product branch consist of origin classes describes its product information (product name, place of manufacturing, name of distributor and other information), i.e. the class origin of og phone product branch maybe the different type of the phones that the entity support (Samsung, Apple and other types). This module is mainly used to count the words in the text of reviews that related to the branch entity to calculate a percentage result.

**Rashmi & Vivekanand et al.** (9) designed a model of spam detection reviews on an amazon dataset about a movies, the proposed technique used in this paper is similar to the technique used in (8). After getting the review, clustering is used into groups generating an ARFF file based on a set of features, which in turn will enter the classifiers. This paper divided the work into two classification tasks, the sentiment classification and the spam classification, the sentiment classifier depends on semantic orientation calculation algorithm using an NLP parser, the spam classifier depends not only on the text of the review but also on the rating of that review. After finishing classification, the classified fake reviews enter more calculations using the association rule to calculate a threshold for the support count if the review get a value higher than that threshold it will be considered as a brand spam review.

**2.1.4 Hybrid method of spam reviews detection**

**Algur et al.** (10)followed Language Modeling Approach for Consumer Review Spam Detection to detect fake reviews (type1) and non-related reviews (type3). shown an approach of spam detecting the spam reviews by measuring its trustworthiness. The researchers used in their work the pre-processing techniques like POS, stop word removal and stemming the data, they also used their own POS tagger based on the word-net lexicon and the availability of Word-net API. and they used the unsupervised probabilistic language model to detect the un-truth review spam (type 1) and a supervised classification to detect the non-review spam (type 3). For the un-truth reviews they design a computational model using KL (Kullback-Leibler) divergence. For (type 3) they identify features which were used in detecting the web spam such as Lexical, Stylistic and Syntactical features. Then they used the logistic regression and SVM (Support Vector Machine).

**2.2 Spammers detection**

**2.2.1 Based on review text**

**Shojaee et al** (11)developed a stylometric-based model, this study focused on the writing style of the reviewers to detect spam reviews. Stylometric features can be divided into lexical and syntactic features. Total length or average length of reviews, ratio of the characters in word and ratio of a non-character or number in reviews is an example of lexical features. While the frequency of function words and punctuation is an example of syntactic features. They used the same gold standard dataset that created and used by **Ott et al.** (2). Then they extracted 234 of stylometric-features (lexical and syntactic), and they used WEKA machine learning tool for conducting their experiments. Different algorithm was adopted, such as SVM and Naive Bayes models, they separated the data based on features so they put the lexical and syntactic features and calculate the F-measure for this model, and also calculate the F-measure for the combined dataset to find the was 84% which is higher than using separated dataset.

**Li et al.** (12) focused on heuristic features. They used data from Epinions.com and labeled them manually. They considered that reviews with high rank and helpful comments is more to be trustworthy, while the others review with low helpful are suspected to be spam. They ignored reviews from anonymous users and removed the duplicate reviews. Then they used supervised learning such as naive Bayes, SVM and logistic regression to detect the spams. Two groups of features were extracted from the data, the first group was content similarity features, sentiment features and content features from text (unigrams & bigrams), while the second group is more reviewers related features (reviewer behavior & profile features). The result of the model shown that the naive Bayes model achieved much better accuracy using only behavior features. It scored 0.583 F-measure using all features which is not good. But they conclude from the model that the behavioral features are very important since the accuracy is dropped significantly excluding them.

**Mukherjee & Kumar et al.** (13) This paper avails other’s work, criticize other’s work, and improve the techniques used by them, this paper depends on a unsupervised Bayesian machine learning technique, that clusters the reviewers into two classes spammer and non-spammer based on a set of features that can expose spammer through studying the behavior of reviewers, and another set talking about detecting the fake reviews taking into consideration the author of this review and his behavior and history that was explored in the previous set of features. Almost all the features in this paper depends on pre-defined thresholds, these thresholds is found using a small labeled dataset (spammer/ non spammer) and this represent some super visioning in the model, but as they consider themselves working on an unsupervised technique, they exclude this dataset from the final evaluation. The novel ASM (Author spamicity model) which gives the authors a degree in range in [0,1] of being spamming, the authors who are taking into consideration are only the reviewers who have 3 or more reviews, the results evaluation is done using a supervised classification depending on the results of the unsupervised spam opinion model without the need to manually labeling the data. The main dataset used in the ASM model is a real-life data on a manufactured product on amazon, this large dataset consists of about (50K reviewers with 1 million reviews), with all the needed metadata used in the features proposed in this paper

**2.2.2 Based on reviewer account metadata**

**Fei et al.** (14) focused on the behavior of reviewer for detecting the spam reviews in short time burst. So that burst can be due to the sudden popularity of the product or it is a spam. They built a network of reviewers which appears in many different bursts, then to represent this network into a graph named as (Markov Random Field). Then they used a statistical model that helps the classification of reviewers as spammers or not. As we mentioned that the researchers depend on the behavioral features of the reviewers such as the rating deviation and bursts of reviews, these features are normalized as a range [0, 1]. As the previous model the authors ignore single reviews. Since we need more than one review for one reviewer to build statistically reasonable reviewer behavior. This model achieved 77.6% accuracy, however this model is used to detect the fake reviews in short bursts.

**Lim et al** (15)focused on the behavior of reviewers to detect the fake reviews. This model assumed that the spammers target certain products or brand to make the most impact. They also depend in their model on the rating behavior since the spammer will give a product rating different from the average rating of the all reviews. The data on which this model used are collected from Amazon that is contains the review text with rating in rage [0,1]. Then the reviews were filtered by keep the reviews of users who are active or at least wrote three reviews. Then different models were designed, i.e., targeting product (TP), targeting group (TG), general rating deviation (GD) and early rating deviation(ED). All these models based on many spammer behaviors since each model producing a separate score, then to combined these scores to calculate the final spam score. However, they focused in their model on few products believed from them that the spammers will monitor only few products. And then they will write a spam review when the time is suitable to change the rating. So, they rely on targeting products in short period since the rating will be completely different from the others, and they will try to affect the other reviewer’s opinions so early.

**Liy et al** (16) working with Dianping, first large-scale analysis of restaurant reviews filtered by Dianping fake review filtering system in this model. Since Dianping is a company that developed an efficient fake review filtering system. This study works on large scale real life restaurant reviews. Since it is differing than the previous works by the data volume since it is shared by Dianping with user’s ID which is around 6 million reviews, also it differs by the data richness compared to other datasets since it contains of users IP address, users profile which are allows authors to create more features used in machine learning models and also with feature novelty since this is the first to give ideas of temporal and spatial features, such as reviews, user profiles and IPs of users. For this proposed method the authors implement some novel temporal and spatial features for supervised spam reviews detection. Temporal Patterns show some longitudinal studies along the time dimension. For example, spammers are active during specific days and less active in weekends and this what the studies of temporal patterns shows. However Spatial patterns is to study the arrangement of spammers on earth and includes the space between them. So, these authors rely on these features on their model of spammer detection.

**Xie et al.** (17) focused in the singleton reviews in constraint of other researchers who works in multiple reviews per reviewer, since more than 90 percent of the reviewers write only one review. Also, the author noticed that the size of singleton review is big compared to non-singleton reviews. So, these reviews have more impacts at the rating of product. Then they used to extract a relationship between the volume of singleton class and the rating of the store claimed that if there is an intense increase in the volume of singleton and sharp either high or low of the store rating, we conclude that the spammers attempt to control the store rating early to get more impact. For each window time the authors calculate three metrics that is the ratio of singleton reviews with average rating and average number of reviews. Next to find a multidimensional time series using the previous metrics by fitting the time curve on each dimension. Then to apply the LCS (longest common substring) algorithm. Then on each point of the three dimension the authors measure the spam burst, then they design a multi-scale anomaly detection algorithm. Finally, they used human judges to define data as spam or not and compared the labeled dataset by human by the output of this algorithm, to find that the accuracy was 75.86%.

**2.2.3 Based on graph features**

**Wang et al.** (18)proposed one of the first studies that used graph representation in fake review and reviewer detection in their methodology. This approach is different than the other approaches, since they don't used the text

of review, also it is able to detect more ambiguous cases of spam reviews activities. The authors study training to detect kinds of spammers who can manipulate their behavior to act just like genuine reviewers and think that they cannot be detected by other techniques. Author propose a novel based method of heterogeneous review graph to detect the relationships between the reviewers, store and reviews. They represent each element previews as graph, since relationships between these nodes can be extracted by explore the interactions between these nodes. Then to identify the suspected reviewer. They also developed a mathematical method to identify the trustiness of reviewers, how much the reviews are honest and the reliability of stores.

**Mukherjee et al.** (19)designed a multi-edge graph to detect spam reviewers, which is based on extract the relationship between reviews, reviewers and store. The considered that is very normal for reviewers to have multiple reviewers due to the different purchases, hence it is expected for stores to have duplicate reviews since different stores may provide same product. The heterogeneous (multi-edge) graph contains mainly three nodes: reviewer, review and store, since each node is linked to another nodes. Since the reviewer’s node is linked to the reviews which them wrote, and each of these reviews is linked to a store linked. Moreover, each node is attached to set of features such as each reviewer has number of reviews, average rating and rating for each review. They believed that if there similar rating supports the same idea so there is no conflict of opinions, and there is a conflict if the rating is different. Three metrics are introduced in these processes which is trustiness of reviewers, reliability of the store and honesty of review, this three metrics affected by each other such as disease in honesty of review will affect the reviewer’s trustworthiness, while the trustworthy reviewers will have amount of honest reviews. They calculate these three metrics measurements and used them to rank reviewers, reviews and stores, and according to the evaluation results, the reviews/reviewers with highest ranking are most likely to be spam/spammer.

**Akoglu et al.** (20)proposed a fast and efficient method Fraud Eagle to detect a fraudsters and fake reviews. This model has several advantages since it used the network effect between products and reviewers and don't base on the text of reviews. The dataset mainly consists of set of reviewers, products and reviews, since each review is written by a giving reviewer about a particular product with a star rating as integer in range of [1-5]. Whilst the nodes (reviews, reviewers and products) are connected to each other as network graph and each link between two nodes associated with a rating review. The proposed model consists of mainly two steps, firstly to give a score to the reviews and reviews for spam detection purposes. Then to grouping to be more representative and helpful. This method adopted unsupervised learning techniques that used unlabeled datasets. Also, it is a scalable for larger datasets with linearly grows time with network size. The authors hired propagation-based algorithm that in rule employed the advantage of the network effect to detect the relations between nodes. Then to infers the classification for the reviewers as spammer or honest, reviews as fake or true and products as good or bad by the giving score to each node and updating the score iteratively. Also, to propagate it to the neighbor’s nodes via the connectors of the graph.

**Chapter III**

**Proposed work**

In the recent years, online reviews played an important role in the purchasing decisions, so many of researchers as discussed in chapter 2 proposed many methods to identify these fake reviews. Many of them depend in their approach in the opinion mining and emotion using text analysis which is includes extracting lexical features (n-gram, POS, content similarity and semantic similarity) from the text itself to determine if the review is spam or not. While the others researchers used to identify fake reviews by extracting spammer behavioral features. However, a few of them used to combine these two types of features together in their model.

The first approach of detecting fake reviews restricted by the content of the review only, moreover It is possible for a company to employ different people to write fake opinions that cannot be detected by text analysis since each spammer is had a specific style in writing. Moreover, it should take in consideration that as the spammer can reuse their reviews they also can re-change the reviews entirely so not all of these fake reviews will be detected.

The second approach is using the behavioral of spammer by analyzing the attitude of reviewers to detect fake reviews. This done by extracting behavioral features that is related to the spammer itself (IP address, rating of the product, window time, number of posted reviews …). Whereas behavioral model is used to detect spammer or a group of spammers who work together in the same area using same IP, same account or same product. These methods are a good indicator if the review that written by a given reviewer is spam or not but it is not necessary. However, they ignore that the reviewers not necessarily to have a duplicate or near duplicate reviews since they can change their attitude and style of writing in order to increase the spam effectiveness and also to deceive the exist filtering algorithms. Also, spammers not necessarily to post their reviews in different time bursts with different machines so can avoid the proposed techniques that depends on time features and IP address.

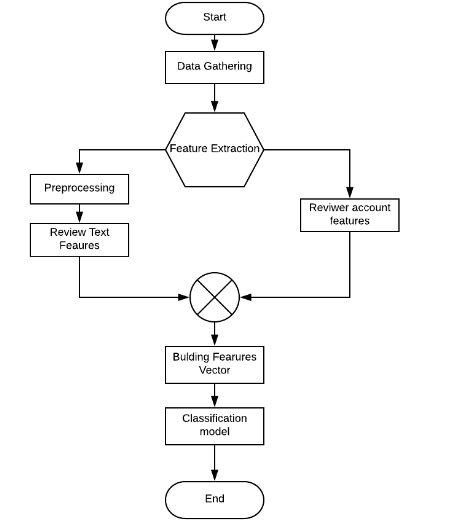
**3.1 Notation**

Let Rh= {r1, r2, …} be a set of reviews for a particular product or place H, where r representing the review object which consist of 2 sets r• =<Reviews Attributes,User Attribute>**.**

**Review Attributes.** Each review consist of a set of 3 attributes, Review Attributes={Text ,Rating ,Date}, where text  represent the review text which consist of set of words and denoted as ‘Rtext’, Rating represent the star rating of the review which values are in the range from zero to five, high rating give the reader an idea of how the reviews are helpful, and denoted as Rrate’, and the Date represents the date and time when the review was posted as form of year-month-dayand denoted as ‘Rdate.

**User Attributes.** Each user attributes consist of a set of 3 attributes, User Attributes= {ID, Name, AvgRating, Count}, Where ID represent the unique user id that maps to the user since it is helpful in identifying the duplicate submissions from same id since it is forbidden to have a conflict opinion in the same product donated as RId. Name representing the name of reviewer as a group of characters donated as ‘RName’. AvgRating represent the average overall rating of a particular user in all brands that give a general idea about the rating behavior of a reviewer donated as ‘Rrating’. And Count which represent the all numbers of reviews that related to a particular user donated as ‘Rcount’.

**3.\* Proposed work**



**3.2 Feature extraction**

This section explains the features that will be used in this project, and mention at least one method to observe the related feature with the mathematical model to find it, all depending on the notations section:

**3.2.1 Text Features**

Text features are the features that can be extracted from the text of reviews. Such as duplicate and sentiment similarity and other Stylistic features which includes a lexical attributes (e.g. word length distribution and special characters frequencies) and structural attributes (e.g. vocabulary richness measures, digit n-grams and function words).

**3.2.1.1 Duplicate**

As many spammers tend to write many reviews on the same product, and this process is time consuming and the imagination of spammers is limited, so they tend to copy the reviews or part of it, due to this content similarity is one of the main features in exposing spamming activities since this feature can measure content similarity of the reviews which is used to detect the duplicate reviews. If there is a review is similar to another review in content, we can consider these two reviews as considered to be duplicate of each other’s and thus can be considered as spam. However, this feature will not be able to detect unique spam reviews since it is designed to detect the duplicate reviews.

As we want to compute similarity of two reviews so we need to identify the similarities between their words. So, the first step is to represent the review as bag of word (vector space model). Given two review texts and, each review can be modeled as a n-dimensional vector where each dimension is representing one word using TF/IDF to get and as two vectors which is one of the ways to uniform the term frequencies is to weight a term by the inverse of document frequency. TF/IDF is given by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.1.1.1)** |

Where

Cosine similarity is measure that calculate the cosine of the angle of two non-zero vectors, where the outcome is neatly bounded in [0, 1]. Their cosine similarity that gives can be formulated as

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.1.1.2)** |

As this feature based on the text of reviews so the text need to be cleaned. Preprocessing is the process of cleaning the data from unwanted elements and also to normalize the reviews since the same words sometimes written in different form to perform accurately and get more efficiency in the proposed model. Preprocessing techniques is summarized in the following common steps.

Figure 3.2.1.1.1: Review preprocessing stages.

Cosine similarity gives a high advantage of being simple to compute since there is many tools used to compute the cosine similarity between two documents, also many researchers as Salton and Leask (21) who used the lexical matching using cosine similarity get good results in the smart IR system. However, cosine similarity was able to identify the few duplicates of exact matches (e.g. ‘NIKON d7100 is a great camera’ and ‘nikon-d7100 is a great camera’) but couldn’t able to find the similarity between a little different duplicate since it failed to identify the synonyms of the words (e.g. ‘hotel was awesome’ and ‘hotel was great’). Thus, we will use a semantic feature to be able to identify more difficult duplicates.

One of the cons of content similarity in finding similar sentences that have the same meaning (contains synonym), this problem can be solved using a semantic feature that depends on the WordNet semantic dictionary, this feature is considered as a natural language processing feature (NLP), that measures the sentence similarity (i.e. meaning similarity) of two reviews.

“” WordNet is a lexical database for English Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations””

The lexical similarity approach will be used in to implement this feature, this process has 6 processing steps are shown in the next Figure,

Figure 3

We will calculate each of semantic and duplicate features between two reviews separately, and then combine them into one feature according to the following equation.

|  |  |  |
| --- | --- | --- |
|  |  | Eq. (\*\*) |
|  |  |  |

Where r1,r2 are the reviews and is a constant that will be set experimentally.

**3.2.1.3 Length of the review**

Sometimes non-spammers prefer to write short review texts as people tend to skip long reviews. According to the previous work Akoglu et al (20), they argued that the average length of non-spammers should fall in a specific range. This feature can be measured by the following equation.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.1.3.1)** |

If then it may be an indicator of a review spam.

**3.2.1.4 Advertisement**

Spammers sometimes use to embed a URL-link within the review that not related to the product itself maybe to promote the other site, as the URL-link consist of two components which is base and path. For example for the next URL-link “[https://www.amazon.com/  
ExerciseThought-Extra-Graphic-T-Shirt/dp/B0174B8HNO/](https://www.amazon.com/ExerciseThought-Extra-Graphic-T-Shirt/dp/B0174B8HNO/)”, the base URL is [www.amazon.com](https://www.amazon.com/Exercise-Thought-Extra-Graphic-T-Shirt/dp/B0174B8HNO/) while the path of the site is “[Exercise-Thought-Extra-Graphic-T-Shirt/dp/B0174B8HNO/](https://www.amazon.com/Exercise-Thought-Extra-Graphic-T-Shirt/dp/B0174B8HNO/)”. We consider that if the base URL that mentioned in the review it is not matched with the site base URL then it should be a good indicator that the review is spam. (e.g. if the URL-link of the site is “<https://www.amazon.com/Exercise-Thought-Extra-Graphic-T-Shirt/dp/B0174B8HNO/>” while the URL-link mentioned is “[https://www.aliexpress.com/item/detail/32818868551.html?spm=  
2114.10010108.1000023.10.41f9505dtYRfaZ](https://www.aliexpress.com/item/detail/32818868551.html?spm=2114.10010108.1000023.10.41f9505dtYRfaZ)” then the noticed is these two base URL’s are not matched thus this type of review maybe a spam review, since we want to use Amazon dataset so any review contains a URL with base address that doesn’t matched “<https://www.amazon.com/>” may be a spam review. This type of spams is an indicator that the review is spam of non-related type since these types of reviews does not give helpfulness to the peoples.

**3.2.1.5 Brand or Price mention**

Spam review can be specified as review on brand only, although it can be a true review, but it doesn't helpful for the others who see the review since it is not related to the product but talks about the brand of the reviewed product or the price of the product or maybe about the seller of the product. This feature can give the review a score (BScore; which is brand score) by each times the brand name, price or to check if there are key words such as manufacturing, seller, delivery, etc. are mentioned in the review text. Bscore can be calculated by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (1)** |

Where Wordi is the special word which donates that the review on brand or not, n is the number of special words, then to check If Bscore>Bthreshold where Bthreshold represent the review on brand threshold, then it is good indicator that the review is spam review of review on brand type.

**3.2.2 Spammer features**

**3.2.2.1 Normalized maximum number of reviews**

Analyzing the behavior for the spammers they note that posting many reviews from the same user within time specified period is good indication for spamming activity because spammers usually tend to write more than one fake review on the same product or different products so the number of reviews for him became larger. Since we have the posting time for the review we can extract Maximum Number of Reviews (MNR) as follow:

(3.2.2.1)

|  |  |
| --- | --- |
|  | **Eq. (3.2.2.1.1)** |

So, when the result is greater than or equal predefined threshold then the reviewer may be considered as spammer.

**3.2.2.2 Extreme Rating**

Spammers usually tend to give extremely high or low rating to the products in order to promote. This feature alone is not sufficient to decide that the reviewer is spammer so this feature combined with other features to help indicating spammers. This feature is extracted from the rating on the product as follow:

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.2.2.1)** |

Since if the output of the above function is 1 so the reviewer may be a spammer else no indicator that the reviewer is spammer.

**3.2.2.3 Helpfulness rating for the review**

The websites provide the opportunity for users to vote for other reviews from other people by choose if it is helpful or not then automatically calculate the helpfulness rating for the review and this is good indicator that the reviewer is spammer or not. Helpfulness Rating (HR) can be calculated by taking the reviews and calculates the average for helpful reviews that he has and compare it with predefined threshold as the equation shows.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.2.3.1)** |

If the result is greater than the threshold it is more likely to be spammer but if it less than the threshold leads to less likely be a spammer.

**3.2.2.3 Account Freshness**

One of the properties that can expose spammers that they don’t use their accounts for a long time. they write reviews in a short period of time. In contrast the genuine users write reviews from time to time. Depending on this, the freshness of the activities of the reviewers’ accounts can be taken into consideration when exposing the spammer reviewers.

The method used for observing such features is the review burstiness, which is the date difference between the first and the last review issued by the same user Account.

For example, if an account issued all its reviews in a period of 3 or 4 months, that is an ordinary activity for a reasonable number of reviews, but let’s say an account takes 15 days only to write all its reviews, this is a footprint of spamming activity, we can define the degree of **review burstiness** (RB) as the date difference between the first and the last review issued by the same user. This method needs a threshold to done the calculations, which is can be set experimentally.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.2.3.1)** |

Where last () is the last date the user u issues a review and first () is the first date the user u issues a review. This feature gives an output in the range of [0,1] which represents the degree of spamicity.

However, there are users who use their account so frequently, hence this feature will give them a high degree of spam and this can affect the final results, but with the other features we provided in our methodology this problem can be overwhelmed.

**3.2.2.2 Reviewing early**

Early reviews are the most important reviews on any product and can be an influential element on people decisions to try the product or not, so the initial reviews affect the sales of the product significantly. So, spammers always try to be among the 0first reviewers for the products.

This feature can be observed by a method called the **Ratio of First Reviews** **(ROFR)** by finding the ratio of the first reviews to the total reviews for the same reviewer and compute it for every user to track the spammer reviewers. ROFR is given by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.2.2.1)** |

Where SFR is the size of the first reviews set that the user u posted while the is the total size of reviews of that user. However, if the dataset provides more than one product, this feature can be helpful.

**3.2.2.4 Rating Deviation Score**

Spammer maybe gives different reviews in different brands; hence spammer will have a rating on the review itself and the average of other overall reviews in different brands. This feature gives the variance between the review rating and average rating of the reviewer, which given by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.2.2.4.1)** |

Where is the brand, is the probability of the reviews on brand divided by the total reviews, is the average rating for the brand, is the average overall rating. This feature is helpful in identifying spammers since having a high deviation on a product with ratio of overall rating reviews is a good indicator for the reviewer to be a spammer.

**3.3 Review Classification**

**3.3.1 Classical Approach**

The first approach of spam detection is used the classical supervised machine learning techniques which requires a labeled training dataset classified as spam and not spam to build the model that will be used later to predict the unlabeled data as testing dataset. Then this labeled dataset used to extract a combination of text features and the reviewer features searching for good result which then represented as feature vectors as a set of 9 features F= {f1, f2, …, f9}. Depending on these features using the classifiers (e.g. SVM, Naive Bayes, KNN, DCT, etc.) we can build and training the model. Hence, applying testing (unlabeled) dataset and comparing the classes given from each classifier with the true classes to calculate the confusion matrix and all accuracy measures.

**3.3.2 Probabilistic Modeling**

Generally, talk, the probabilistic model (Bayesian modeling) is a way to describe the data that can be observed from a system using the mathematics of probability theory (Bayesian rules), to express all forms of uncertainty of that model. Since our system depend on a set of features this set consists of two heterogeneous subsets, a subset of review (text) features f ϵ {DUP, ADV, RLEN} = Ft, and the other subset of the reviewer features f ϵ {MAXR, ER, HLP, AF, RERLY, RDS} = Fr. our system will classify the reviews into one of two classes spam or not spam depending the on the mentioned features set according to the following equation.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.3.2.1)** |

where wt, wr is the weight of the text features and the reviewer features respectively, their sum equal to 1.

As shown in the previous statistical equation, classification of a review will not depend on the two subsets of features in an equally manner, instead it will give each subset a weight that will be calculated using the genetics algorithm.

**3.3.3 Score Based Model**

Score based ranking will give every feature a score or weight according to its importance and effect on classification the reviews. In decision theory the weighted sum model will give every feature in our features set a weight, the following equation describes the weighted sum model criterion,

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (3.3.3.1)** |

Where fi ϵ all features mentioned, wi is the weight of the feature i in the decision of classifying the review as spam or not where where n is the number of features. Again, these weights are calculated and optimized using the genetics algorithm.

**Genetics Algorithm Optimization**

A genetic algorithm is a simulation of the genetic state of a population of individuals, their combined chromosomes represents their genetic state. This algorithm also simulates the operations happen in the genes such as mutation and crossover. Here we will apply the genetic algorithm to optimize the feature weighting in both the probabilistic and the score-based model.

First a set of initial weights is given based on our expectations and knowledge of each feature, then the algorithm processes to find better solutions through an iterative manner. The figure () below shows basic stages of genetics algorithm.

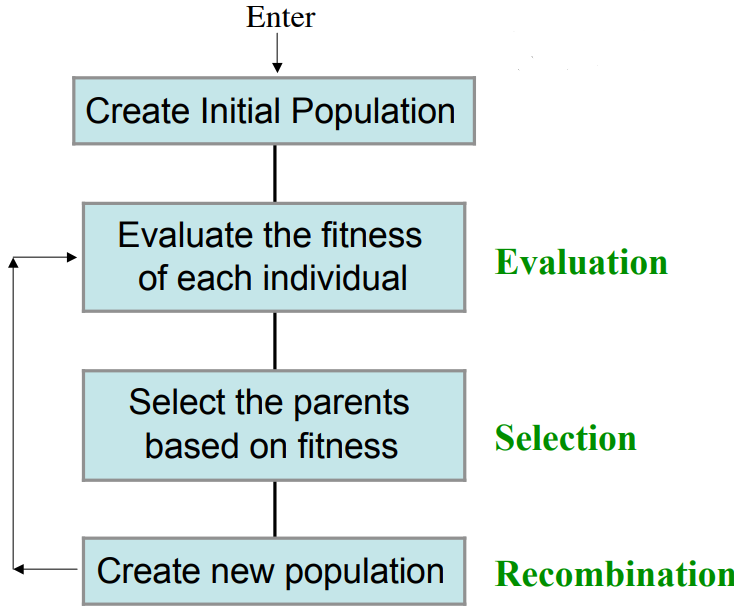


Figure 3.3.3.1.1: Genetics algorithms steps

As the optimization of weights is a genetics problem, we need to formalize the problem; chromosomes are the basis of creation the population. Since each chromosome contains a set of genes. In our case each weight will be represented as gen while the chromosome will be a set of weights as in the following figure ().

|  |  |
| --- | --- |
| Gen1 | Gen2 |
| Weight1 | Weight2 |

Algorithm will start with a set of chromosome as initial population then to evaluate each chromosome by the fitness function which returns a quantitative value, then to select the set of best chromosomes based on the values of fitness function of each, then to apply genetic operators (Crossover and Mutation) one by one to find the most appropriate weights.

In the first stage we must define the parameters that the genetics algorithm will optimize, then encode (e.g. in binary) each one of them as a gene. Suppose after randomly initialize the two weights wt, wr with values 40, 60 respectively, these are percentage values under the value of 100 and since we are using the binary encoding each gene will consist of 7 bits so, the genes form of wt, wr are 0111100 and 0101000 respectively. We will have two chromosomes each one consist of two genes, each gene represent a weight in the probabilistic model, the figure **#####**

Shows the parent chromosome that will be created initially and randomly.

The second stage is to assign a fitness for each chromosome (gene) based on the weight derived from the chromosome, the fitness function should express the goodness of each generation, in our model it will be calculated based on the primacy results we will got experimentally.

The next the perform a recombination using crossover or mutation, to get new generation and select the best of them using the fitness function that we have supposed, as shown in the next figure.

|  |  |
| --- | --- |
|  |  |

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

Parent #1

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

Parent #2

Perform mutation or crossover

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |

Child #1

in this case, the param1 is inherited from parent#1, and param2 is inherited from parent#2 through a mutation recombination operation.

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

Child #2

Figure #####################

برايي هيك بكون احسن التمثيل !

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 | W14 |

Parent #1

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 | W14 |

Parent #2

Perform mutation or crossover

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 | W14 |

Child #1

in this case, the param1 is inherited from parent#1, and param2 is inherited from parent#2 through a mutation recombination operation.

Param1 (wt) param2(wr)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 | W14 |

Child #2

Figure #####################

**4** **Datasets, Tools and Evaluation measures**

**4.1 Dataset**

One of the most difficulties in machine learning problems is to collect and choose the dataset that will be used for training and testing the proposed model, hence we are looking for informative data in our work to extract the desired features to be dependable in our approach of detecting the spams reviews. However, there is another difficulty which is how to collect dataset with the true labels for the model training and testing. There are four sources to get datasets such as Yelp, Amazon, Google My Business API and data sets from previous related work in this field, since most of researches uses these sources to get the data. However, referring to the related works it may easy to have negative reviews (non-fake reviews) since there is sites like Yelp and Dianping that filters the reviews and display just honest reviews and used different ways for data annotation.

Most of the work was trying to create a manually labeled dataset and that what happens with **Li et al. [14]** since they a principled way to have an annotated dataset. They found more others related work about the spam reviews. They then employed 10 college students to annotate 60,000 reviews manually after reading the collected related work to know what the spam and the way to label spam reviews. Each student should independently label the data by determining its spam or not, each review should be labeled by different two student and if there a conflict in labeling then it will be resolved by a third student. **Sun et al. [25]** proposed a synthesis process to generate fake reviews from a pool of truthful reviews, then to measure the performance according to the existing detection algorithm and human readers, they suggested that the fake reviews can be automatically generated by replacing the sentences of a truthful review with other reviews sentences.

**Shojaee et al. [17]** proposed a novel technique to annotate the dataset that collected from amazon.com in this research by providing an information and metadata about reviews and reviewers. The authors developed an online annotation system to improve the accuracy of annotation process. It’s necessary to have a labeling dataset for spam detection approaches to evaluate the proposed method. However, it is difficult to manually annotate the dataset since it is confusing for humans and consuming a lot of time. The authors hire an existing works and proposed some rules to increase the accuracy of labeling dataset. They proposed a method to manually labeling be simultaneously considering clues by design an algorithm of asking questions, 11 questions to detect spam reviews and 5 for spammer detection, since all of these questions are extracted from previous existing work which is selected based of features selection. The algorithm calculates the two number (k, k’). Since if there a clue of spammer detection the algorithm increment k by one. And for every review it calculates a number of spam detection clue and then increment k’ by one for each found clue. Then in case of k’ is greater than a determined threshold(k’) the review will be set to 1 or spam since the algorithm detect the reviewer as spammer. Otherwise to check the if k is greater than a threshold(k). And set the review to spam too. And if the two numbers are less than the related threshold the review will be set to 0.

We have Four available choices for the data set. The first one is Google Api[[2]](#footnote-2), this data set is provided by google for business so you must have business account and pay money for google to get the data set so, we discarded it after trying to get these reviews we found that they provide just the top five reviews about products or places. The second choice is the public data sets from other related work in this field, we contact some persons via email to get the data set but they didn't respond to us so we can’t depend on the datasets they have. The third choice is Yelp data set, this data set provided by Yelp[[3]](#footnote-3) website and it is large enough about (174000) review. This dataset contains the desired metadata we looking for to extract our features from it such as rating, helpful, reviewer, review count, and other attributes. But the main disadvantage that it is not labeled as spam or non-spam which will make an extra step for us to label these data manually or by supervised learning. The last choice is Amazon data provided by Amazon[[4]](#footnote-4) website, it has the required metadata we are looking for and scalable but it is not labeled as spam or not. In general, it was difficult to get labeled data set so we have to choose among these available choices, as a result we determine to work on Amazon data set.

For our data set we will follow a procedure to annotate the data which is first giving every review a score indicates if it is spam or not then determine a threshold for classifying the data in two parts, the first part will be the reviews higher than the threshold which represents non-spam reviews so, it labeled as non-spam reviews. The second part will be the reviews with score less than the threshold for this part of reviews will classify them manually as spam or non-spam.

There is another challenge among with getting labeled data and desired metadata that is the difficulty to have [[5]](#footnote-5)fake reviews since there is sites like Yelp Dianping that filters the reviews so getting fake became harder.

**4.2 Tools:**

Detecting spam reviews is considered a text classification problem and people used different tools for this problem such as Weka, Knime, Orange.

The tool we will used is Weka since it has a big collection of machine learning techniques such as naive Bayes, Decision tree, K-means, KNN, neural networks, etc. Also, this tool has a lot of preprocessing mechanisms for the text such as transformation, loading, and extraction. Moreover, it takes the sentence and represent it in featured vector using TF/IDF. it’s an open source and implemented in java platform.

The other tool we will use is Stanford NLP which provides a lot of processing techniques for the text including part of speech tagging(POS), removing stop words, removing punctuation marks, and named entity recognizer(NER). This tool is open source and supports many languages including the English (in which we work). Also, its written in java. So, we will use this tool for preprocessing for the data before sending it to Weka.

The input for the Weka is a set of labeled data set. Firstly, we will do preprocessing for the data such as removing stop words in the Stanford NLP tool. Then, we represent each review by a word vector using TF/IDF in Weka. Here, the data is ready to be classified so, we will use four different algorithms to classify these data which are Naive Bayes, Neural Network, KNN, and Decision Tree. we will see which classifier gives the best result so we can depend on it for the classification work in this project.

**4.3 Evaluation Framework**

As we develop a solution to solve problem using machine learning then we need to choose metrics to evaluate our machine learning model, since these metrics influence how the performance and accuracy of the implemented machine learning algorithms. However, these metrics is standard in the machine learning domain which is precision, recall and f-measure.

**4.3.1 Confusion Matrix**: is an easy way to represent the resulted values from the algorithm and it contains four values as shown in the figure below.

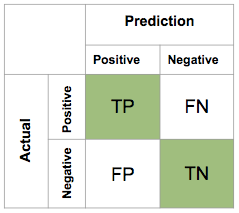


Figure 4.3.1.1: Confusion Matrix

1. **True Positives (TP):** means how the actual class of the data point is classified true as predicted.
2. **True Negatives (TN):** means how the actual class of the data point is classified false as predicted.
3. **False Positives (FP):** means how the actual class of the data point is classified true as predicted while the predicted class is negative.
4. **False Negatives (FN):** means how the actual class of the data point is classified negative as predicted while the predicted class is true.

**4.3.2 Accuracy:** this type of accuracy measurements used to show how many correct classification points related to the all prediction operation. This type gives an overall idea of how the model is accurate. It represented by the following relationship.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (4.3.2.1)** |

**4.3.3 Precision:** measure that shows how many TP operation related to the number of all points that classified as positive (TP and FP). This type of measures is useful in critical cases. Precision is represented by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (4.3.3.1)** |

**4.3.4 Recall:** measure that tell us how many TP classes which TRUE is and classified as TRUE related to the total number of actually TRUE. Recall is represented by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (4.3.4.1)** |

**4.3.5 F-measure:** is a measure which is the harmonic mean of precision and recall which is in the best case 1 and 0 at worst case. The f-measure is given by the following formula.

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (4.3.5.1)** |

**4.3.6 ROC curve:** the receiver operating characteristic curve is a plot of the true positive rate(sensitivity) as a function of the false positive rate(specificity) for different cut-off points, every point on the curve represents a (sensitivity, specificity) pair corresponding to specific decision threshold [1]

|  |  |  |
| --- | --- | --- |
|  |  | **Eq. (4.3.6.1)** |
|  |  | **Eq. (4.3.6.1)** |

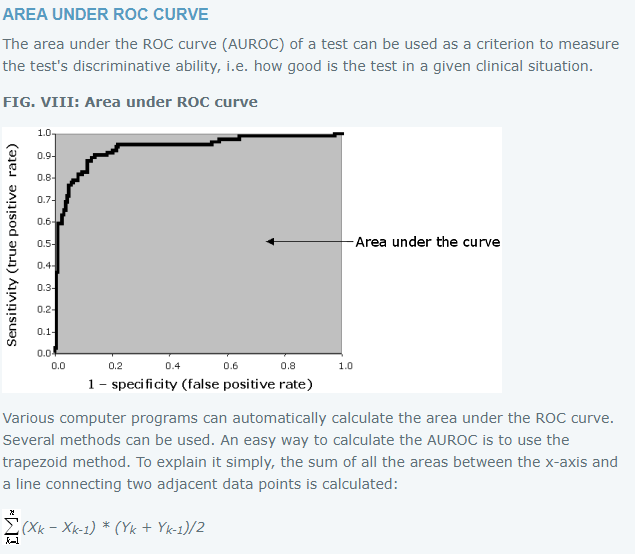
**

Figure 4.3.2.1: ROC curve evaluation measure.

**5 Conclusion**

Due to the increasing of consumer dependence on online reviews, spammers tend to write manually or by proposed review system fake reviews on a product or organization. Since customers’ decisions are affected by the posted review, fake reviews become a dangerous prospect for online reviews. However, customers find that is difficult to identify bad from good products. According to this the posted reviews should be filtered as spam or not spam to be more confident.

In this project, we have focused on identifying spam reviews problem using many features that extracted from the reviews after specify them as text which includes duplicate and semantic similarity and spammers behavior features which includes number of rating, account freshness, reviewing early and rating deviation and other features. As the major challenge in identifying fake reviews is come from how to extract best set of features that will be used in the model. However, these features are very limited. Many researchers who suffers from lack of accuracy depends on their work on textual features and the others depends on the behavioral features while a few of them relied on their work on a combination of these features. We will use a hybrid features to boost the performance and accuracy of our model.

We will depend on three approaches of our spam detection model. Firstly, we will classify reviews as spam depends on the combination of extracted features using classical supervised machine learning techniques (SVM, KNN, Naive Bayes, DCT, etc.). Secondly, using semi-supervised learning depends on probabilistic based on genetics. Finally, using unsupervised learning that depends on score-based optimized by genetics.

# References

1. *Finding unusual review patterns using unexpected rules.* **Nitin Jindal, Bing Liu and Ee-Peng Lim.** 2010, In Proceedings of the 19th ACM Conference on Information and Knowledge Management, pp, pp. 1549-1552.

2. **Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey Hancock.** Finding Deceptive Opinion Spam by Any Stretch of the Imagination. 2011.

3. *Identifying Multiple User ids of the Same.* **Qian T, Liu B.** 2013.

4. *Detecting Opinion Spam and Fake News Using N-gram.* **Ahmed, Hadeer.** 2012.

5. *Detecting Spam Review through Sentiment.* **Zhong, Qingxi Peng and Ming.** 2014.

6. *AN OPINION SPAM ANALYZER FOR PRODUCT REVIEWS USING SUPERVISED MACHINE LEARNING METHOD.* **DAXINI, DAYA L. MEVADA and PROF VIRAJ.** 2015.

7. *Opinion spam recognition method for online reviews using ontological features.* **NGUYEN HOANG LONG, PHAM HOANG TRONG NGHIA and NGO MINH VUONG.** 2014.

8. *Fake Review and Brand Spam Detection using J48 Classifier.* **Sushant Kokate, Bharat Tidke.** s.l. : International Journal of Computer Science and Information Technologies, 2015.

9. *Detection of Fake Review and Brand Spam Using Data Mining Technique.* **Reddy, Miss Rashmi Gomatesh Adike and Prof. Vivekanand.** s.l. : International Journal of Recent Trends in Engineering & Research (IJRTER), 2016.

10. *Conceptual level Similarity Measure based Review Spam Detection.* **Siddu P.Algur, Amit P.Patil, P.S Hiremath and S.Shivashankar.** 2010.

11. *Detecting deceptive reviews using lexical and syntactic features.* **Somayeh Shojaee, Masrah Azrifah Azmi Murad, Azreen Bin Azman, Nurfadhlina Mohd Sharef, Samaneh Nadali.** 2013.

12. *Learning to identify review spam. In IJCAI Proceedings-International Joint Conference on Artificial Intelligence.* **Fangtao Li, Minlie Huang, Yi Yang and Xiaoyan Zhu.** 2011.

13. *Spotting Opinion Spammers using Behavioral Footprints.* **Arjun Mukherjee, Abhinav Kumar, Bing Liu, Junhui Wang, Meichun Hsu, Malu Castellanos, Riddhiman Ghosh.** 2013.

14. *Exploiting Burstiness in Reviews for Review Spammer Detection.* **Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., & Ghosh, R.** 2013.

15. *Detecting product review spammers using rating behaviors.* **EP Lim, VA Nguyen, N Jindal, B Liu.** 2010.

16. *Analyzing and Detecting Opinion Spam on a Large-Scale Dataset via Temporal and Spatial Patterns.* **Huayi Liy, Zhiyuan Cheny, Arjun Mukherjeez, Bing Liuy and Jidong Shao.** 2015.

17. *Review spam detection via time series pattern discovery.* **Sihong Xie, Guan Wang, Shuyang Lin and Philip S. Yu.** 2012.

18. *Review Graph Based Online Store Review Spammer Detection.* **Guan Wang, Sihong Xie, Bing Liu and Philip S. Yu.** 2011.

19. *Spotting fake reviewer groups in consumer reviews.* **Mukherjee, Arjun, Bing Liu, and Natalie Glance.** 2012.

20. *Opinion fraud detection in online reviews by network effects.* **Leman Akoglu, Mary McGlohon, Christos Faloutsos.** 2013.

21. *Computer evaluation of indexing and text processing.* **Lesk, G. Salton and M. E.** 1971.

1. Yelp: <https://www.yelp.com/dataset> [↑](#footnote-ref-1)
2. Google API: <https://developers.google.com/my-business/content/review-data> [↑](#footnote-ref-2)
3. Yelp: <https://www.yelp.com/dataset> [↑](#footnote-ref-3)
4. Amazon: <http://jmcauley.ucsd.edu/data/amazon> [↑](#footnote-ref-4)
5. [↑](#footnote-ref-5)