2013-04-23

Towards Review Spam Detection

Keshavarz-Rahaghi, Fatemeh

UNIVERSITY OF CALGARY

Towards Review Spam Detection

by

Fatemeh Keshavarz-Rahaghi

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

CALGARY, ALBERTA

APRIL, 2013

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled “Towards Review Spam Detection” submitted by Fatemeh Keshavarz Rahaghi in partial fulfilment of the requirements for the degree of MASTER OF SCIENCE IN COMPUTER SCIENCE.

_____

Supervisor, Dr. Reda Alhajj
Department of Computer Science

_____

Dr. Jon Rokne
Department of Computer Science

_____

Dr. Wael Jabr
Haskayne School of Business

_____

Date

Abstract

Nowadays, millions of products and services are available to the public online. Therefore, searching for the best products which targets the individuals' requirements would be difficult as the result of the existence of too many offers. One of the most reliable approaches to choose a product or service is to exploit the experiences of the people who have already tried them, and so have reported almost honest opinions about them. A reviewing system is a place where individuals write their reviews on their experienced products and services, and also benefit from others' reviews. Moreover, companies utilize reviewing systems to apply opinion mining techniques in order to improve their goods or services and to watch their competitors. However, the popularity of the reviewing systems ignites this motivation for some people to enter their fake review to promote some products or defame some others. These review spam should get detected and eliminated in order to prevent misleading potential customers. Opinion mining should be adapted to locate and eliminate potential spam reviews. In this thesis, some review spam detection approaches have been proposed and examined over a sample dataset. The proposed approaches consider the patterns existed in the trends of the reviews, as well as the reviewers' behaviors. The approaches depend on various strategies such as observing abnormal trends, detecting uncommon or suspicious behaviors, investigating group activities, and so on.

Acknowledgements

I would like to sincerely thank my supervisor Prof. Reda Alhajj who has guided me through this thesis as well as all the study and research I have done during these years. I appreciate his kind supports and his high understandings of the situations.

I am also thankful of my dear parents who have always encouraged me and supported me to follow my aspirations in my life.

Finally, I want to thank my dear friends who have been there for me in the hard times I had during these years, and have compassionately helped me. My appreciations to Hessam Zakerzadeh, Seyed Hamed Tabatabaie, Reza Karimpour, Mohamed Alshalalfa, Mohsen Mollanoori, Arash Niknafs, Maryam Elahi, Ala Qabaja, Omar Zarour, Fatemeh Arbab, Seyed Hossein Ahmadinejad, and Soheila Aalami.

To my beloved parents and sister, and all the friends

who supported me during this journey.

Table of Contents

List of Tables

List of Figures and Illustrations

Chapter One: Introduction

1.1 Problem Definition and Motivation

Almost immediately after the advent of the Internet and the Web technology, individuals as well as public and private organizations became interested in marketing their products and services on the Web. Since then, the utilization of the Web for selling products became rapidly pervasive such that nowadays it constitutes a considerable proportion of the Internet usage. Along with advertising on the Web and in addition to the independent sites that allow for product evaluation, vendors have given their audience the authority to evaluate and comment on the merchandise. Having this feature available and in most cases added to the retailing websites, both parties, i.e., producers and consumers can benefit from it. Companies can analyze the provided reviews to learn the strengths and weaknesses of their goods and services from the consumers' points of view, and they can further apply their conclusions in the future to achieve higher levels of satisfaction from consumers, as well as to attract more people towards buying their merchandise. Moreover, they can investigate the purchasers' reviews of the products of the competitor companies to become aware of the general appetite of the consumers: their preferences, priorities, and so on.

Considering the admirable features of the competing products in the purchasers' opinions, producers can improve their own products by adding or magnifying those features. Furthermore by becoming aware of discreditable characteristics of products from rival companies, manufacturers will make sure not to strengthen those

characteristics in their productions or maybe to consider deficiencies before they are realized by the customers. On the other hand, since the ability to review products online is made available for people, they can easily post their comments on their desired products at any time, from any part of the world, and at their own convenience. Consequently, a large number of opinions with a significant variety of ranking can be provided. This also gives at almost no cost the opportunity to competitors to watch how their products have been received by the consumers.

Possessing the sentiments of previous consumers of different interests and expectations, individuals can distinguish their most preferred goods or services, which match their own preferences the most, faster and straighter than the time when such an integrated collection of opinions was not available. Therefore presently, individuals rely extensively on the reviews available online. It means that they make their decision of whether to buy products or not by analyzing the existing opinions on those products. In fact if a potential customer gets a positive overall impression of a product by considering the present sentiments for that product, it is highly probable that he will actually purchase the product. Normally if the percentage of positive opinions is considerable, it is likely that the overall impression will be highly positive. Likewise, if the overall impression is negative, it is less imaginable that vendees buy the product. Again, the overall negative impression can be the result of a great proportion of negative sentiments.

The results of a survey conducted in early 2012 indicate that 51% of the customers have used Internet more than 6 times during a year while 72% of them have the same trust in online reviews as they have in personal recommendations [35].

The substantial importance of online reviews for a vast range of its users from companies to individual purchasers gives this stimulus to organizations to manipulate the overall polarity of the opinions for products. For example, a vendor might put effort to influence the potential customers by tricking them or in other words deceiving them. This is possible by turning the overall sense of the company's own products positive by hiring some people to post several positive comments for those products in order to attract more potential purchasers toward buying them. On the other hand, a merchant might be willing to ruin the overall sensation of the products offered by competing companies, and thus pay some individuals to write their unrealistic negative feelings about those products. Consequently, an alteration in the general feeling of a product can have extensive impacts on its supplying company by affecting its credit, and eventually by affecting the amount of interest the company can gain through selling that product. Consequently, there is a need for an automated process that could analyze online comments with the hope to identify and highlight the unrealistic one which have been introduced merely to affect the overall opinion whether negatively or positively. This motivated for the work described in this thesis. I have developed some novel techniques for identifying fake negative and positive opinions.

1.2 Contributions

The intention of this research is to distinguish the opinions posted about products to intentionally change the overall sentiment of the products from those comments written to simply reflect the genuine thoughts of real purchasers telling their experiences of the

products. However, this is not a trivial task to accomplish since those who are getting paid to write desirable comments for their employers do their best to publish comments which resemble the ordinary comments throughout applying various tricks, so that they will not get easily caught and be eliminated from the system. Therefore, it is a complicated job to discover such comments and this job cannot be done manually by the readers of those comments especially when the publishers of the posts are very professional in their assigned duties. As a result, an automated system is needed to detect those unrealistic posts. This system needs to employ several techniques towards determining misleading opinions, i.e., it should consider the problem from different points of view in order to be able to detect even the professionally written untrue comments. The reason of introducing multiple methods is that the spammers utilize different approaches to mislead the reviewing systems, and therefore they need to be treated in different ways.

Then the solution for the problem of discovering untruthful posts should take all the comments as well as all possible approaches into account.

In this research, I have proposed some approaches to solve the problem described above; each approach considers the problem from a unique point of view. I have investigated the sets of comments on the products, as well as the behaviors of the individuals to detect unreliable posts.

1.3 Overview of the Proposed Framework and the Dataset

The introduced framework mainly contains the implementation of five proposed approaches for detecting fake reviews. The first approach considers the patterns in the polarity of the reviews for each product. Its aim is to indicate the unusual intervals in the trend of the polarity of each product. The second, third, and forth approaches focus on the behaviors of the reviewers. The second one detects those reviewers who have many extreme-rated reviews while the third approach points out the reviewers whose reviews are extreme-rated on their targeted products, and are rated with a rating similar to the average rating of the corresponding product for their non-targeted products. The forth approach concentrates on the density of every reviewer's reviews over time, and marks the reviewers who have many reviews in a few days interval. Finally, the fifth approach focuses on finding the spammers who work together as a group by taking the outlier reviews, whose ratings noticeably deviate from the average rating, into account. In other words, this approach detects the groups which their members review the same products together while each time one of the members gives an outlier review, and the others give reviews with ratings close to the average rating.

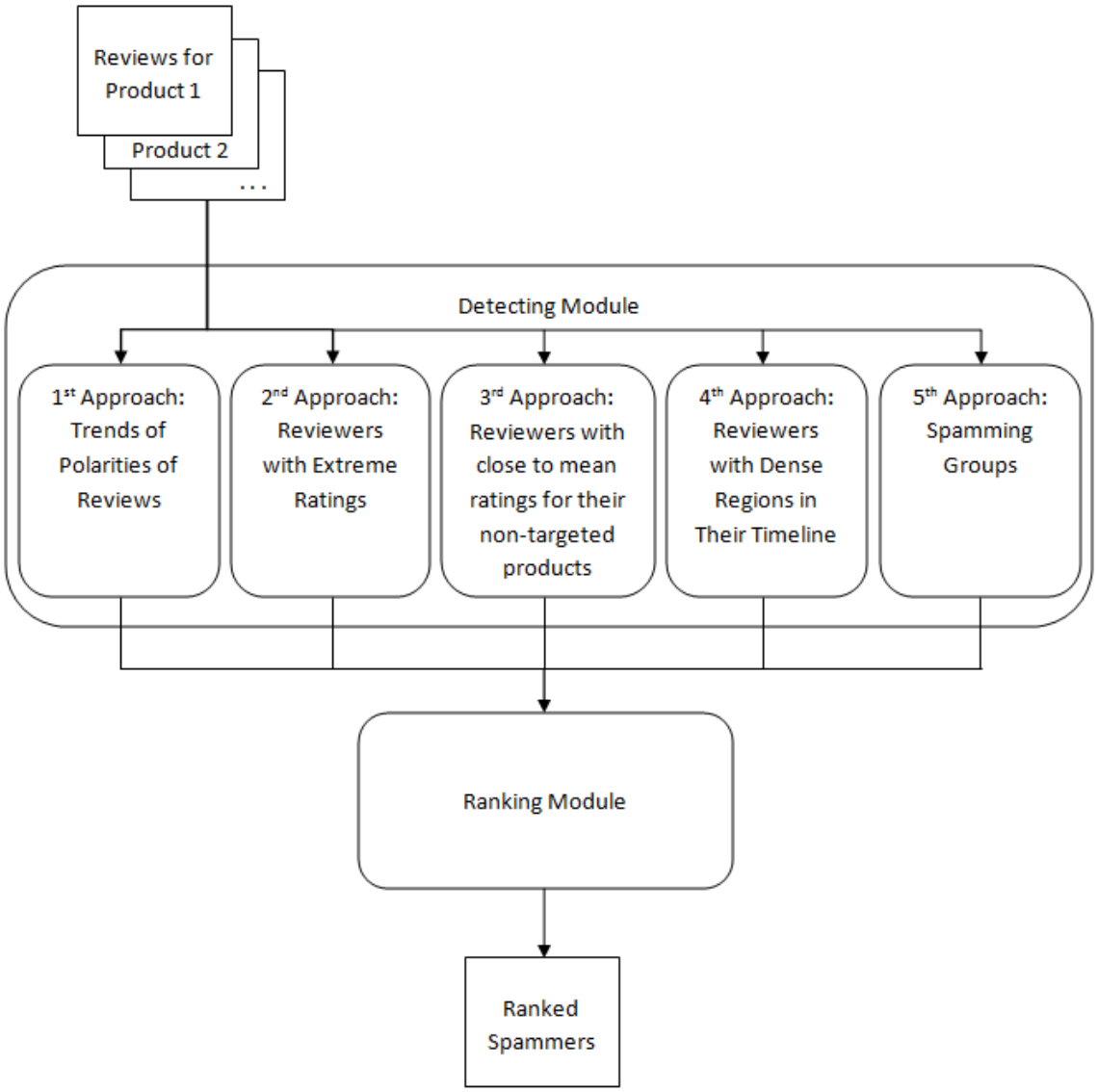
Each of the approaches has a different point of view, and therefore they detect spammers with different attitudes. Consequently, if a reviewer has been detected by a greater number of the approaches, it is more likely that the reviewer is an actual spammer. As a result, a ranking module has been utilized in our framework. This module ranks the detected spammers descendingly, according to the number of times they have been detected by the approaches. Hence, if a spammer has been found in the results of all the

approaches, it will appear at the top. Finally, the list of the ranked spammers is saved as the output of our proposed framework.

The introduced framework is domain-independent, and therefore can be applied on any set of reviews from any reviewing system.

The block diagram given in figure 1-1 illustrates an abstract overview of our framework.

Figure 1-1 A block diagram illustrating the proposed framework



The dataset, on which I have applied my proposed framework, has been taken from [3]. After removing the reviews with missing values, I came up to a set of 1,131,482 reviews. Each review contains the information about the ID of the reviewed product, the ID of the reviewer, the date when the review is posted, and the rating given by the reviewer to the product. There exist 474524 products, as well as 27217 reviewers in the dataset. The ratings are given in a range from 1 to 5. The products which have been reviewed are books, music, DVD, and industry manufactured products, like electronics, computers, etc.

A snapshot of the dataset is shown in figure 1-2.

Figure 1-2 A snapshot of the dataset

Product ID	Reviewer ID	Posting Date	Rating
B000077VQC	A1004AX2J2HXGL	October 19, 2003	5.0
B000005X1J	A1004AX2J2HXGL	October 17, 2003	4.0
B00000JZXJ	A1004AX2J2HXGL	October 17, 2003	5.0
B000002P4L	A1004AX2J2HXGL	October 15, 2003	5.0
B00000K19E	A1004AX2J2HXGL	October 13, 2003	2.0
B000024KBA	A1004AX2J2HXGL	October 12, 2003	5.0
B000066EX9	A1004AX2J2HXGL	October 9, 2003	5.0
B000005JAC	A1004AX2J2HXGL	October 6, 2003	5.0
B00008J4P5	A1004AX2J2HXGL	October 5, 2003	5.0
B00004Z4WX	A1004AX2J2HXGL	April 4, 2002	5.0
B00003CX9S	A1004AX2J2HXGL	December 12, 2001	5.0
B000004BPD	A1004AX2J2HXGL	November 25, 2001	4.0
B00000276F	A100TW8FZECWD6	March 30, 2005	5.0
394756444	A100TW8FZECWD6	June 6, 2004	5.0
B00009KO14	A100TW8FZECWD6	March 13, 2004	3.0
B000096I8G	A100TW8FZECWD6	August 21, 2003	4.0
B00008DDVV	A100TW8FZECWD6	June 26, 2003	5.0
B00006FDR8	A100TW8FZECWD6	September 23, 2002	5.0
1885173172	A100TW8FZECWD6	February 16, 2002	3.0
425181685	A100TW8FZECWD6	November 4, 2001	2.0
441003745	A100TW8FZECWD6	July 4, 2001	4.0
6305019681	A107I6YPYHLZIC	April 11, 2003	5.0
B0000004AX	A107I6YPYHLZIC	March 18, 2003	5.0
B00006EXEH	A107I6YPYHLZIC	January 20, 2003	4.0
B000068QEN	A107I6YPYHLZIC	October 18, 2002	4.0
B00006ALA4	A107I6YPYHLZIC	October 4, 2002	4.0
973110910	A107O8UATN67M8	January 10, 2005	5.0
1590710029	A107O8UATN67M8	September 28, 2004	5.0
1566633877	A107O8UATN67M8	September 28, 2004	5.0
B00000FC7M	A107O8UATN67M8	October 17, 2002	4.0
6302969352	A107O8UATN67M8	October 16, 2002	5.0
B00000JSJ6	A107O8UATN67M8	October 2, 2002	3.0
B000031EGO	A107O8UATN67M8	October 2, 2002	4.0
195014766	A107O8UATN67M8	September 6, 2002	5.0
B00003CXZ4	A107O8UATN67M8	August 15, 2002	4.0
60510277	A1084J87F6KKDO	April 10, 2004	3.0
345464869	A1084J87F6KKDO	October 15, 2003	5.0
767912020	A1084J87F6KKDO	April 8, 2003	3.0
60256656	A1084J87F6KKDO	May 24, 2001	5.0
525446222	A1084J87F6KKDO	May 24, 2001	5.0
694013218	A1084J87F6KKDO	July 4, 2000	2.0
1555610730	A1084J87F6KKDO	February 19, 2000	3.0
486275892	A1087DECRN5UDU	April 19, 2000	5.0
048626792X	A1087DECRN5UDU	April 19, 2000	5.0
486298531	A1087DECRN5UDU	April 19, 2000	5.0
486253368	A1087DECRN5UDU	April 19, 2000	5.0

1.4 Thesis Organization

The rest of the thesis is organized as follows. In Chapter Two, I have studied the existing approaches towards revealing the unrealistic comments. Chapter Three gives a general overview of my proposed ideas for tackling the problem. I have divided my suggested methods into three categories; methods detecting fake posts by considering the relations between all comments on the products, methods detecting fake posts by investigating the behaviors of the individuals, and methods detecting fake posts by discovering the groups of people working together to write those posts. Moreover, the approaches have been clarified by some examples in this chapter. In Chapter Four, the detailed algorithms of the proposed methods are illustrated. The implementations of the algorithms are also given in this chapter. Furthermore, some samples of the results have been demonstrated in chapter Four. Finally, Chapter Five concludes the thesis, and explains the future work directions.

Chapter Two: Background

The concept of review spam has drawn considerable attention recently. Different researchers have tried to look at this problem from various points of view. One of the first approaches toward this issue has been introduced by Jindal and Liu [1, 2, 3]. They have claimed that Email spam and Web spam as well as Recommender system's spam have been studied widely so far while much less investigation has been done on review spam. In their approach, first they have categorized review spam into three main groups:

1. False Opinions; which are classified into *Positive Spam Review*: an undeserving positive opinion to promote a product, and *Negative Spam Review*: a malicious negative review to damage the reputation of a product.
2. Reviews on Brands Only; which are about the brands of the products rather than the products themselves, and thus are highly biased.
3. Non-reviews; which contain no opinions at all. They can be advertisements for the same product, different products, or the same product from different seller. They can also be Questions, answers, comments on other reviews, or just some random text unrelated to the product.

Then, Jindal and Liu described their spam detection strategy in three main steps;

1. Detecting duplicate and near-duplicate reviews;

2. Detecting type two and type three review spam; and

3. Finally Detecting type one spam using the results of the first step.

As the first step, they have found duplicate and near-duplicate reviews from the same userID on different products, from different userIDs on the same product, and from different userIDs on different products. They have used such reviews as the base point of their review detection method. However, they count duplicate reviews from the same userID on the same product as a mistake in clicking the submit button more than once, or a correction of mistakes in former reviews by the same person.

For the second step, the authors have considered finding the second and the third types of review spam as a classification problem with two class labels; *spam* and *non-spam*. They have used logistic regression so that a probability of being spam has been assigned to every review, and the reviews could be ranked by their probability and thus there was no need to strictly remove any review as spam. In this way, all the reviews are labelled with values indicating how probable it is that they are spam. Therefore, instead of removing some reviews as definite spam, and keep the rest as definite non-spam, reviews with different probabilities can be treated in different manners. To build their classification model, they have considered various features from reviews, reviewers, and products.

In the last step, they have claimed that positive spam reviews on products with positive average rating, and negative spam reviews on products with negative average rating are not that harmful. Therefore, they have focused on finding outlier spam reviews at this phase. To build their classification model, they have taken duplicate reviews found in the first step as the *spam* class members, and all other reviews as *non-spam*. Also, they have utilized the same features as those of the second step. Then, the authors have applied the above classification model to find outlier reviews. They have declared that their model finds reviews of negative deviation well enough, but not those of positive deviation.

Some other observations they have had are as follow: People who write many comments on one brand, where almost all of them are negative, are likely to be spammers; Only-reviews, each of which has been posted as the one and only review of its corresponding product, are likely to be spam; Top-ranked reviewers (those who get many helpful feedbacks) are likely to be spammers since they write too many reviews, some of which might be only-reviews or reviews with high deviations. Spam reviews can get good feedbacks so the number of these feedbacks can't give us much information about a review being spam or not. Those products with high-rank sales are less likely to receive spam reviews since it's hard to damage the reputation of a high selling product by writing just a few numbers of spam reviews; and finally spammers who want to promote bad products don't give those products very high ratings in order not to get caught easily.

Another idea discussed in [4] focuses on reviewers' behaviors to find spam reviews. The authors have described two major aspects of reviewers' behaviors to model; "Spammers may target specific products or product groups in order to maximize their impact.", and "Spammers tend to deviate from the other reviewers in their ratings of products." The authors have claimed that in order to find spam reviews, it is better to concentrate on reviewers rather than reviews since the amount of information that can be gained from the trends of reviewers' behaviors is much more than the information obtained from reviews. Then, they have proposed that spammers are more likely to write spam reviews for one or a few target products. For those reviewers who have targeted one product, they have come up to these results:

If the person has written several reviews with different ratings, he might not be a spammer, and it might be the case of him changing his opinion on the product. However,

if he has written reviews with similar rating on the same product, or reviews with resembling content, he is likely to be a spammer. Regarding the reviewers who have aimed a group of products, the authors have applied clustering over each reviewer's reviews which were given in a short time span. Their conclusion was that large clusters with high ratings have a high probability of containing positive spam reviews, and large clusters with low ratings are more probable to include negative spam reviews. Finally, they have ranked the reviewers based on the results of the above approaches by giving a probability of being spammer to each of them.

The authors of [5] have tried to discover the spammer groups which are groups of people writing together promoting or defaming reviews on products. They have considered these groups so harmful since they can establish trends of opinions on products. They have assumed a group of reviewers who have repeatedly written their reviews on the same products as a possible spammer group if any of the following cases has arisen; they have written their reviews in a short time span, the ratings of their reviews for a single product have deviated from the rest of the ratings for that product, the contents of their reviews have been exact copies of each other, they have given their reviews on a product just after the product became available for criticizing, they have formed a great proportion of the reviewers of a product, they have made up a large group (since it is less likely that they have become a group by chance), they have worked together on a considerable number of products. After investigating the above cases for the groups of reviewers, the authors have ranked the discovered groups regarding the probability of being a spammer group for each group.

It has been argued in [6] that spam reviews can be detected by distinguishing unusual behaviors from reviewers. The authors have aimed to design a domain-independent framework to address this type of problems by discovering unexpected rules from data utilizing association rule mining. They have claimed that short rules are of more interest than long ones since they don't cover data appropriately. Thus, they have only considered rules with one condition or two conditions. They have defined their rule using attributes such as reviewerID, productID, and brand of the products to make the left-hand side of the rules; and they used positive, negative, and neutral (according to the ratings of the reviews) as their class labels to make the right-hand side of the rules. They have started with creating one-condition rules, and followed by calculating values of confidence unexpectedness as well as support unexpectedness for those rules. They have applied the same thing for two-condition rules. So, for example if there is a rule saying: "If the productID is "B00028HBKM" then the class label is positive", and the unexpectedness values calculated for this rule are high, it means that highly probably this rule is a case of spam, i.e. for the reviews about this product, if the given ratings by its reviewers are positive, it is highly likely that those reviews are spam. Finally, they have ranked the provided rules based on their computed unexpectedness values. The more unexpected a rule is, the more likely the review related to that rule is a spam review.

Authors of [7] have designed a framework for operating review mining, and they have declared that a significant part of their system is the review spam identifier module because the miner system shouldn't be misled by fake reviews. As their approach to detect review spam, first they have manually labeled a random subset of data as spam or non-spam to create the training and testing sets of their classification model. They have

claimed that spam reviews are not so helpful, and considered reviews with low helpfulness ratings as spam. Therefore to create the spam dataset, they have first grouped the reviews based on their helpfulness value into three categories: top, middle, and low helpful sets. Then, they have picked a number of reviews from each category, and extracted some information about them. Afterwards, they have asked human judges to mark those reviews as spam or non-spam. Using the above labeled data, the authors have applied several classification methods, and recognized Naïve Bayes Classifier as the best supervised approach. Having the observation that most of the reviews of a spammer are spam reviews, they have introduced a two-view semi-supervised method, a bootstrapping method, to discover review spam. This approach first tries to find spammers based on the so far found spam reviews, utilizing the manually labeled reviews. It then predicts more spam reviews based on the so far found spammers. These steps are repeated iteratively till all reviews are labeled as spam or non-spam.

It has been claimed in [8] that having a limited number of labeled data for review spam classification is not enough and so will give poor results. Thus, the authors have suggested to utilize a semi-supervised classification model called "Two-view Transductive SVM" which takes advantage of *Transductive SVM* as well as *multi-view learning*. In TSVM the unlabeled data are used as well. Transduction is a kind of reasoning in which a limited number of cases are provided out of some seed cases, in contrary to induction where a general rule is extracted out of the seed cases. In the case of SVM, being transductive means the unlabeled data are also used in the training phase instead of inducing class labels for the testing data using the model trained by only the training data as in traditional SVM. Additionally, the authors have provided two different

views of data to boost up the classification results. The outcome was a two-view transductive SVM, which defines two TSVM classifiers for both labeled and unlabeled data, and trains them simultaneously. After completion of the training phase, a weighting system can be applied to integrate the outcomes of the two classifiers. For instance as one of their experiments, they have defined two types of features for building the classification model; lexical features, and formal features, where each group of features has been considered as one of the views, and so a TSVM has been trained for each.

The authors of [9] have looked at the problem of detecting spam reviews from three different points of view; "a text categorization task", "an instance of psycholinguistic deception detection", and "a problem of genre identification". Furthermore, they have proposed a different classifier for each point of view, so the features for each classification problem were to be determined beforehand. The features to be used in the "genre identification" classifier have been selected based on part-of-speech tags.

In the "psycholinguistic deception detection" approach, the authors have utilized the results of a well-known text analysis tool named LIWC (Linguistic Inquiry and Word Count) to obtain the features for their classifier. The outcome of LIWC is a number of psychological dimensions, including linguistic processes (such as average number of words per sentence), social and emotional processes, personal concerns (like money and religion), and so on. Considering each dimension extracted from LIWC as a feature, they have built their classifier. For their "text categorization" method, they have used UNIGRAM, BIGRAM, and TRIGRAM feature sets provided from both content and context. An n-gram model represents sequences_, e.g., of words_ by predicting the probability of each word having all its previous words in the sequence. An n-gram based

text classification tries to deal with textual errors by using the flexibility provided by n-grams to deliver more reliable classification results [12].

Having features for all the three methods in hand, the authors have applied Naïve Bayes as well as SVM classifiers which have been claimed to lead to reasonable results. Finally, they have employed a 5-fold cross validation to emphasize the better performance of their proposed approaches comparing to human judges, with the combination of LIWC and BIGRAM as the best model for review spam detection.

It has been argued in [10] that in reality most of the reviewers write only one review, namely singleton review. It has been claimed that singleton reviews have been ignored in previous studies on review spam detection. Since a spammer intends to change the polarity of the rating of the targeted product and meanwhile he doesn't want to get easily caught, he writes his reviews using several identities in a short time span. Thus, his reviews appear as singleton reviews which need to be brought into consideration.

The authors have declared that if a considerable number of singleton reviews have been found on a product written in a short time interval, and the rating of that product has been manipulated as well, there would be a high possibility of the existence of such spammers with several fake identities. Therefore, the authors have tried to find the correlation between the fluctuations in ratings and the masses of reviews. First, they have divided time into consecutive time windows and have found the average rating, the number of reviews, and the ratio of singleton reviews for each time interval. Then, they have plotted one different diagram with time as the x-axis for each group of values (i.e., one diagram for average rating, one for the number of reviews, and so on). Afterwards, they have detected the bursting regions in each of the diagrams by matching them to a template of

bursting patterns, and pinpointed the top K bursting points in each of the three diagrams. Finally, if a time window has contained bursting points from all the three diagrams, it would be inferred that a spam attack has occurred during then. However, their approach cannot be utilized for our study since all of the reviewers in our case have at least a number of reviews, more than one for sure. Thus, this method cannot reveal any of the spammers for our data.

The authors of [11] have asserted that finding spammer groups is noticeably important in review spam detection as a group of reviewers can significantly influence the overall sense of a product. Moreover, it is much easier to detect groups of spammers rather than individual spammers. It has been argued in the paper that considering only the reviews contents or reviewers' abnormal behaviors wouldn't be of much help in finding spam reviews since the substantial number of people in the spammer groups can easily distribute the spamming process among themselves to compensate abnormal behaviors.

Since the spammers in a group write their reviews on the same group of products, they can readily get discovered using approaches like Frequent Itemset Mining. However, not all the groups found are spammers since some reviewers might have fallen into same groups as a result of resembling tastes, and so on.

Proposed by Agrawal et al. [14] in market-basket analysis, Frequent Itemset Mining is the task of detecting the sets of items which frequently occur together in customers' transactions. The idea is that there is no need to investigate all possible combinations of items as frequent itemsets. In fact, if an itemset has been proved to be infrequent, all of its supersets will be infrequent, and do not need to be further examined.

In this research to discover candidate spammer groups, the authors have mapped reviewers to items, and each set of reviewers who have commented on each product as a transaction. So, each product represented one transaction. By applying frequent itemset mining, those groups of reviewers who have reviewed same products together many times would be detected. On the other hand, the authors have claimed that since no gold standard dataset exists for review spam detection, the best way to have a labeled dataset is to have it labeled manually by some domain experts. Thus, they have asked their human judges to classify reviews as "spam", "non-spam", and "borderline". But in contrast with previously manually labeling efforts, they provided the judges with some metadata about reviews. Then, they have assigned a "spamicity" value to each group based on the manually determined labels, and they ranked groups based on that value. Afterwards, to define their proposed model, they have introduced two sets of spam behavior indicators to be used as their needed features; group behavior, and individual behavior.

Group behavior indicators were as follow:

(1) Group Time Window: is a value between 0 and 1 calculated according to the time interval in which group members have posted their reviews on some products. The smaller the time interval is, the closer the GTW is to one.

(2) Group Deviation: is a value indicating the deviation of the average rating of the reviews produced by group members from the average rating of the reviews published by others. The more deviated the group's average rating, the closer the GD is to four.

(3) Group Content Similarity; indicates the degree of similarity among the contents of the reviews written by the group members. The more similar the reviews are, the higher the GCS is since it is more common for spammers to copy each other's reviews.

(4) Group Member Content Similarity: shows how similar are a single reviewer's reviews to one another. If a considerable number of reviewers of a group happen to have similar reviews, it is more likely that the group is a spammer group. The more resembling the individuals' reviews are, the closer the GMCS is to one.

(5) Group Early Time Frame: is a value between 0 and 1 representing how early the group members have posted their reviews. The earlier their reviews have been published, the closer the GETF to one since spammers tend to write their reviews as the first ones on a product to control the polarity of opinions about the targeted product.

(6) Group Size Ratio: shows the average of the ratio of the number of group members to the total number of reviewers for each product. The closer the GSR is to one, the higher is the probability of the group to be a spammer group.

(7) Group Size: indicates how large the group is. The greater the GS is, the more likely the group members are spammers since it is less probable that they have formed the group by chance.

(8) Group Support Count: denotes how many products the group members have reviewed together. The higher the GSUP is, the more probable the group is a spammer group since its members work on many same products together.

Individual spam behavior indicators were as follow:

1. Individual Rating Deviation: indicates the deviation of the reviewer's rating from the average rating.

2. Individual Content Similarity: shows the similarity of the contents of a reviewer's multiple reviews on one product.

3. Individual Early Time Frame: represents how early the reviewer's review has been posted.

4. Individual Member Coupling in a group: shows how close to other group members the reviewer usually posts his reviews chronologically.

Having the above features as well as the manually labelled data, a classification model can be utilized to label reviewers' groups as spam or non-spam. However, it couldn't be done in this problem since their data instances, i.e., groups were not independent as they might have shared members, and also group behavior would not represent individuals accurately. Therefore, the authors have proposed their own approach which models the "group spam - products", "member spam - products", and "group spam – member spam" relations.

Groups and products are related in the sense of their spamicity values. Since the spamicity of products depends on the spamicity of the reviewers groups who have commented on them, and vice versa. Members and products, as well as groups and members are related according to the same concept. Consequently, they have calculated the spamicity for each pair in a bootstrapping manner using their related features. Thus, for each entity_ products, groups, and members_ two distinct values have been computed. The authors have defined a ranking algorithm to perform the above calculation and rank the existing groups based on their spamicity. Finally through their experiments, they have proved that their method outperforms the supervised classification and regression algorithms.

It has been declared in [13] that previous studies are not sufficient for spam detection since they have not considered some subtle facts of real world. For instance, one person's reviews might look alike because each person has a unique writing style, or spammers can also write good reviews because they might be real customers of some other competitors, or they might not be spammers any more. Thus, the authors have proposed to utilize a tripartite graph with reviews, reviewers, and stores as its nodes, to model the relations among these three entities. Furthermore, they have defined three concepts; the trustworthiness of reviewers, the honesty of reviews, and the reliability of stores, which are affected by each other. Moreover, they have introduced a recursive method to calculate the three concepts utilizing the defined graph. To perform the above tasks, first they have explained each concept. The trustworthiness of a reviewer is a value in the range (-1, 1) denoting how trustworthy he is.

A reviewer's trustworthiness_ how much a reviewer can be trusted_ can be extracted using the honesty of all his reviews. The authors have calculated this value for each reviewer by taking into consideration that a reviewer's trustworthiness does not depend on the number of his reviews but on their honesty. Moreover, the trustworthiness increases more rapidly when the number of reviews with higher honesty values is smaller, and it rises more gradually when the number of such reviews is larger. The honesty of a review is a score between -1 and 1 indicating how honest that review is. It can be examined through the reliability of the store to which it is related, as well as the consistency of the review with its surrounding reviews, which are of the same rating and within a time interval around the time when the review has been published. The reliability of a store implies its quality through a value in the range (-1, 1) and can be

determined by considering the reviewers' trustworthiness of the store. The same ideas of computing the trustworthiness are applied here except instead of the honesty of reviews, reviewers' trustworthiness should be utilized. As the next step, the authors have proposed an iterative algorithm to find the honesty, trustworthiness, and reliability values using the relations extracted from the graph. Having the candidate spammers as the results of the above processes, they have asked human judges to distinguish the spammers while they have provided the judges with some metadata about the candidates' relations with other reviewers, stores, and the Internet. Thus to label spammers, the judges had this mindset that if most of the times a reviewer's rating differs from the other reviewers' ratings, the store's pre-assessed rating, or the general idea about the store provided by Web search engines, that reviewer is highly suspicious. Finally, the authors have compared their approach with previous ones and have claimed their approach have detected those types of spammers who could not be discovered utilizing the previous methods.

In general, one of the problems with most of the above approaches is that they employ classification to solve the problem of review spam detection. However, the issue with utilizing classification in this area is that the training sets are usually created manually by human judges as a result of the absence of gold-standard data in this field. Consequently, the accuracy of the results would be affected. To talk specifically about the approaches, [7] have based their classification method on the helpfulness scores of the reviews which are not reliable since they might be spammed as well. [9] has used the content of the reviews, and therefore applies natural language processing. The problem with it is that the expert spammers can write their fake reviews in such a way that they resemble ordinary reviews and so would not get detected by utilizing natural language processing.

In [6] the aim is to detect unusual behaviors using association rule mining. They have assigned an unexpectedness score to each rule consisted of a review ID, a reviewer ID, and a brand. The drawback of this approach is that even if a reviewer has changed his opinion or if his opinion is different from the normal case, this approach would detect him as a spammer. The suggested approach in [13] is acceptable but its problem is that it considers the judgment of human judges into account to come up to the final list of the spammers. [10] targets singleton reviews which are given in a short time span and has affected the overall ratings of their corresponding products. The issue here is that there is no singleton review in many datasets such as ours. Moreover, the reviews which are given in the short time intervals, as well as those which affect the ratings have been studied in our framework. [4] has suggested to consider reviewers' behaviors rather than review since they give more information. This concept has been utilized in our framework as well. It has been proposed in [5] and [11] to focus on detecting spammer groups since they have more influence on the ratings, and they try to hide their spamming activity by distributing it among themselves. These are the reasons why discovering spammer groups has also been included in our proposed framework.

Two of the closely related approaches are mentioned below. Then, we proceed to identify the contributions we make above and beyond these approaches. I should refer to those introduced in [4] and [11]. The important idea suggested in [4] which I used in my framework is focusing on the reviewers' behaviors rather than the reviews since much more information can be extracted. However, they consider only two suspicious activities from the reviewers. While in my framework, I have added three more approaches to this category. The suggested idea in [11] is to detect spammer groups rather than individuals

which again has been utilized in my framework. The difference is that they have applied frequent itemset mining to find groups, and then used human judgement to classify the detected groups as spammer or not. The problem here would be the inaccuracy of the result as it is dependent on human feelings, whereas in my proposed framework I have tried to recognize suspicious activities, and to rank the groups based on their probability of being spammers.

Our proposed framework attempts to compensate the drawbacks of the above approaches by looking at the problem of review spam detection from different points of view, and by mining the patterns which exist among reviews, as well as among the reviewers to reveal the spamming activities.

Chapter Three: An Overview of the Proposed Approaches towards Review Spam Detection

Nowadays, many people intending to purchase products or services surf the Internet to search for them to first evaluate those products or services by receiving the opinions of others who have previously experienced them. Moreover, the individuals as well as the companies selling the products or offering the services utilize the online available opinions to have an estimation of the general feelings about their products or services, and to know their strengths and weaknesses to improve them in the future. The potential purchasers tend to seek others' opinions but they need to extract those opinions from the enormous amount of data available online. However, this is a difficult task to accomplish. Different people from different parts of the world with different cultures, different levels of education, and different understanding of the offered goods and services have different expectations and so would likely publish entirely different opinions. Thus, it would be a hard task for individuals to search among all those types of reviews and find the ones which match the most with their interests and priorities. As a result, a systematic approach for this problem has been introduced known as *opinion mining* aka *sentiment analysis*.

Sentiment analysis first tries to extract objects, and their features on which the individuals' opinions are given. *Objects* are the targets of the provided comments, and their *features* are whether their subcomponents or their characteristics. Furthermore, sentiment analysis needs to specify the *opinion holders* who are the owners of the

comments, as well as the *orientation* or the *polarity* of the opinions, which means the opinions being positive, neutral, or negative [16, 17].

There are two types of opinions to be identified and detected by opinion mining systems; *direct opinions* which are the feelings of the individuals about various single products, and *comparative opinions* in which the opinion holders compare two or more products according to their interests and priorities [16]. In the case of direct opinions, opinion mining usually classifies the documents each containing a person's opinion into either positive or negative classes. Some sentiment analysts have tried to detect the orientations of opinions of the given comments by utilizing supervised [18, 21, 26] or unsupervised learning [19, 24]. However even in the texts provided by opinion holders, not all the sentences contain opinions. Therefore, the sentiment analysis system needs to mine the *opinionated sentences* out of the given text.

Opinionated sentences are the ones consisting of explicit or implicit opinions [16]. As a result, some other researchers consider the problem of opinion mining at the sentence level, i.e., they examine each sentence first to assess if it is subjective [22, 23], and then to detect its polarity [20, 25, 26], even though, the sentence-level opinion mining has its own drawbacks. For instance, assume a reviewer has a negative sentiment about a product but there exist some positive sentences in his posted review while those sentences are actually referring to some other products he has mentioned in his review earlier. As an example, a reviewer might say: "I am using this cell phone for about two weeks but I should say it is far away from what I expected. I was satisfied with my last cell phone from this company. It is fast and light. So I thought the new one must be even better." It can be understood from the whole review that the person has a negative

opinion about the reviewed product but some sentences in the review contain positive sentiment.

In the case of comparative opinions, the approach and the intention are different. The mining system needs first to detect the comparative sentences which usually contain some equative (e.g. as … as), comparative, or superlative adjectives or adverbs [28]. Then, it would be able to distinguish the preferred item among all [27, 29].

However, this extensive usage of online *reviews*, i.e., people's opinions posted on the Internet, and their key roles in the process of the potential purchasers' decision-making on whether to buy their intended products or services or not, as well as their significant importance in the supplier companies' assessments of themselves, result in the reviews to possess undeniable impacts on determining the amount of financial benefits and reputations gained by the companies. This fact might give some companies the motivation to try to make the general opinions derived from online reviews of their products and services more positive. Therefore, they might pay some pseudo-reviewers called *spammers* to write undeserving positive reviews on their own products [5, 7] whereas to publish defaming negative reviews on the competing companies' products [3]. Besides, some other reviewers criticize their targeted products while they are too emotional about the products to do so, and hence they write unrealistic reviews on the products.

These fake reviews are known as *review spam*, which should not be mistakenly mixed with email spam and web spam. Email spam implies the irrelevant emails undesirably received by account holders which usually contain advertisements to promote various products or services [32, 33], while Web spam denotes the malicious activities performed

by the owners of the Web pages to bring the rank of their pages higher in the lists of the results of searches [30, 31].

Recently, considerable attention has been drawn towards review spam detection; some of the attention has been discussed in chapter two. The studies in the area can be divided into two major categories: approaches utilizing natural language processing, these methods try to detect suspicious reviews by examining the contents of the reviews separately; and approaches deploying the status of the reviews among others, the relations between reviews, and the interactions between reviewers to discover suspicious activities.

In this thesis, I have concentrated on proposing approaches of type two for two reasons. Firstly, because the type-one methods have been vastly studied previously, and secondly since the professional spammers put their best effort to create their spam reviews such that they resemble the genuine reviews [3] to a great extent, and so they cannot be easily distinguished from honest reviews by applying natural language processing as a stand-alone mechanism. However, natural language processing can be used together with other approaches to give more accurate results.

In the rest of this chapter, I have explained my proposed approaches for review spam detection in broad terms. I have categorized my ideas into three main groups. The first group focuses on the approaches which investigate reviews of each product separately, and try to find suspicious reviews by extracting the patterns or trends of the reviews. The approaches of the second group consider reviews of different products but from the same reviewer. The intent of these methods is to discover suspicious reviewers by examining their behaviors in general and over time. The last group contains the approaches which

regard all the reviews of all the reviewers together. Their goal is to detect the groups of spammers who work together to change the overall opinion of one or more products since they believe their impact would be much more affective while acting as groups.

3.1 Investigating the Trends of Reviews for the Products

The review spam detection approaches which are discussed in this section aim to consider the problem from the perspective of the products. Thus, in the coming approaches, an effort has been put to extract the trends and patterns from the reviews belonging to each product, and moreover to detect suspicious activities through them. The reviewers who have been involved in noticeable number of the detected suspicious activities would be regarded as spammers.

3.1.1 Statistics of the Reviews with Positive, Neutral, and Negative Ratings

Rationally, quality products usually receive positive reviews, whereas the products of poor qualities receive negative reviews in most cases. Since the products with good qualities make more of their consumers satisfied compared to those with unpleasant qualities, it is more likely that they receive higher number of positive-rated reviews. Therefore, the better the quality of a product is, the higher the percentage of its positive-rated reviews. Conversely, a considerable proportion of reviews of a shoddy or lousy product are negative-rated since they make more of their purchasers disappointed. Therefore, for each product, the percentages of its reviews with positive, neutral, and negative ratings are almost consistent during its lifetime since these percentages reflect the overall opinions on the product which are directly dependant on its quality. However, some external causes/factors might affect the general opinion about a product during its

lifetime. For example, a new alternative product becomes available to compete with it, and so the opinions about the product tend to move toward positive or negative, according to the strengths and weaknesses of both products. Nevertheless, these changes in the trends of the ratios over time are not sudden fluctuations, and so in most of the lifetime of the products the ratios of positive and negative opinions remains almost consistent, or follows a steady increase or decrease. Consequently, if the reviews of each product are sorted in a timeline, for most of the selected intervals in the timeline the proportions of positive, neutral, and negative rated reviews should be approximately the same as those proportions calculated from all the reviews of the product.

Sorting the reviews of a product in a timeline means putting the reviews in a chronological ascending order with the earliest published review as the first review in the line, and the latest published review as the last review in the line. If some interval is detected in which the ratios are noticeably different from those of other intervals, i.e. if there is a sudden fluctuation in the ratios computed for that interval, there is a probability that some spamming activities have been done in that time interval. Thereupon, the reviewers whose reviews have fallen into that interval, and moreover the ratings of their reviews are in contrast with the common polarity of other reviews, are labelled as candidate spammers. To explain better, if the ratio of positive rated reviews in that interval is lower than usual, the negative rated reviews in the interval are contemplated to be suspicious since their existence have led to a sudden drop in the percentage of positive rated reviews. So, the owners of such reviews are treated as candidate spammers. In opposite, if the ratio of positive rated reviews is higher than usual, the positive rated reviews are counted as suspicious. The same concept applies when the ratio of negative

rated reviews deviates from the usual case. Thus, if the ratio of negative rated reviews in the interval is lower than usual, the positive rated reviews of that interval are announced as suspicious reviews, and if that ratio is higher than usual, the negative rated reviews are regarded as suspicious.

For my experiment, every time I select an interval of reviews and eliminate that interval from calculation. It means that when I have talked about an interval, I have considered all the other reviews which are not within the initial interval.

As an example Table 3-1 shows the calculated ratios of positive, neutral, and negative reviews for all the sliding windows over the product with the ID of "014029628X".

Table 3-1 Ratios of positive, neutral, and negative reviews for all the sliding windows for a product

Interval	**Percentage of Positive Reviews**	**Percentage of Neutral Reviews**	**Percentage of Negative Reviews**
1st (the earliest reviews)	0.83	0.12	0.04
2nd	0.83	0.12	0.04
3rd	0.83	0.12	0.04
4th	0.83	0.12	0.04
5th	0.83	0.12	0.04
6th	0.79	0.16	0.04
7th	0.79	0.16	0.04
8th	0.79	0.16	0.04
9th	0.83	0.16	0.0
10th	0.83	0.16	0.0
11th	0.83	0.16	0.0
12th	0.83	0.16	0.0
13th	0.83	0.16	0.0
14th	0.87	0.12	0.0

Interval	Percentage of Positive Reviews	Percentage of Neutral Reviews	Percentage of Negative Reviews
15th	0.87	0.12	0.0
16th	0.83	0.12	0.04
17th	0.83	0.12	0.04
18th	0.83	0.12	0.04
19th	0.83	0.12	0.04
20th	0.87	0.08	0.04
21st	0.83	0.12	0.04
22nd	0.83	0.12	0.04
23rd	0.83	0.12	0.04
24th	0.83	0.12	0.04
25th	0.87	0.08	0.04

It can be seen in the above table that the least frequent value is 0.87 which means the reviews in its corresponding interval can be counted as suspicious. Moreover, it is a ratio of positive reviews, and is higher than the usual case. Therefore, the negative-rated reviews in that interval should get labelled as candidate review spam.

After discovering all suspicious reviews by investigating the timelines of all the products, the reviewers of the picked reviews are determined. As mentioned earlier, happening in an unusual interval is not a sufficient reason for a reviewer to be considered as a spammer since he/she might have given a review with an unusual rating because of many possible external or personal motives. Therefore, only those reviewers whose reviews are detected as candidate spam several times will be announced as the spotted spammers.

3.2 Analyzing Reviewers Behaviors

In this section, those approaches toward review spam detection are introduced which try to investigate the reviewers' behaviors while posting their comments on products. Observing reviewers' behaviors refers to examining the reviewers' manners to detect their commenting habits, extracting the patterns they follow in commenting, if any pattern exists, and analyzing them to discover the reviewers' spamming activities. The reason of proposing the reviewer-centric approaches is that for many products there do not exist sufficient numbers of reviews from which a pattern can be extracted. On the other hand, most of the reviewers have acceptable numbers of reviews to be used to provide behavioral trends and apply further analyses to detect abnormal activities [4, 10]. Specially, the suspicious reviewers would publish larger numbers of reviews compared to other reviewers as each of them is expected to provide a minimum number of reviews to accomplish his/her job.

3.2.1 Reviewers with Extreme Review Ratings

One of the suspicious behaviors of the reviewers can be described by considering the ratings of their reviews. There is an intuition saying people in general are not completely satisfied or totally unsatisfied by their consumed products since no perfect product exists in real world. This means if individuals are contented by a product, probably there are still few facts about it which make them unhappy. On the other hand, if some products do not operate as their consumers have expected, it does not necessarily mean that those products are malfunctioned in all their aspects. The rationale behind this fact is that if a product was useless through all its features, no one would be interested in purchasing that

product, and consequently that product would not be produced any more. Therefore, at least few positive points must exist about each available product. Although a few number of reviewers might exist who have reviewed their targeted products emotionally, and therefore have declared their complete satisfaction by not mentioning even a single drawback of the products, or announced their complete dissatisfaction by not giving even a single advantage of the products, most of the reviewers usually publish various combinations of positive and negative opinions in their reviews.

Perceiving the above statement, we find out that the normal reviews will not usually contain extreme ratings since the owners of those reviews will not give full positive or full negative scores to their targeted products. Hence, it is less common for ordinary reviewers to publish many reviews with the lowest or highest possible ratings. As a result, this approach aims to discover those reviewers a considerable percentage of whose reviews are of such extreme-rated reviews, and then to announce them as potential spammers. For example, consider the reviewer with the ID of "A1CY6RGVEG9XOL" all of whose reviews for all his/her reviewed products are shown in Table 3-2. Although he/she has published one review with the rating of 3, he has posted all the other twenty one reviews with the extreme ratings, i.e., 1 and 5. Hence, the majority of his reviews are extreme-rated and so he is very likely to be a spammer.

Table 3-2 All the reviews posted by a reviewer who has many reviews with extreme ratings

Review	Product ID	Review Time	Review Rating
#1	B00000DXQY	December 6, 2005	5.0
#2	B0000AKOL8	December 6, 2005	5.0
#3	B0009NDDDY	December 6, 2005	5.0
#4	B000005BHA	December 2, 2005	1.0
#5	B000006UP5	December 2, 2005	5.0
#6	B00000IYC9	November 29, 2005	5.0
#7	000718381X	November 10, 2005	5.0
#8	B000BJ7D96	November 8, 2005	5.0
#9	B000006UMF	November 2, 2005	5.0
#10	B000006UOM	November 1, 2005	5.0
#11	B0000AKOLJ	November 1, 2005	5.0
#12	B0000TPADS	October 31, 2005	1.0
#13	B00003Q56T	October 17, 2005	5.0
#14	B0007Z9R7A	October 5, 2005	5.0
#15	B000067AS5	October 3, 2005	1.0
#16	B0000046O5	October 3, 2005	5.0
#17	B0000AA8V2	September 23, 2005	5.0
#18	B0002PUHGU	September 23, 2005	5.0
#19	B000042OFF	September 8, 2005	5.0
#20	6305428352	March 14, 2005	5.0
#21	B00017HWM6	December 14, 2004	3.0
#22	0516235877	March 15, 2004	1.0

3.2.2 Spammers with Close to Mean Ratings for Non-targeted Products

The idea of shilling attacks to the recommender systems, which has been discussed in [15], can be mapped to the review spam area to explain one of the spammers' behaviors in order not to get easily detected.

A recommender system gathers people's conjunctive preferences on various items, and aggregates them with each individual's preferences to pick some items from all to recommend to interested people where the suggested items are more likely to be of their interest [24]. However, production companies are enthusiastic to make their sales higher by having the recommendation systems suggesting their manufactured items more often. This fact makes a motivation for them to mislead the system by injecting false information in it. Furthermore, competing companies might want to make the sales rates of each other less than what they really deserve. Thus, they are also likely to misguide the system in the opposite way. Both kinds of the above activities, which aim to intentionally influence the correct knowledge of the existing items in the system, are called shilling attacks to the recommendation systems [15].

The concept of shilling attack in the recommendation systems is so close to the concept of review spam in the online reviewing systems. Therefore, the behaviors of the shilling attackers can be extracted from the recommendation systems domain and then examined in the review spam detection realm to discover review spammers. For instance, the authors of [15] have explained one of the designs of shilling attacks as follows. The idea is that the attackers divide the items into two groups; those items which are their targets of attacks, and the rest of the items. They treat these two types of items in two different ways. For each of the items, the attackers do not intend to spam, they consider a normal

distribution the mean of which is equal to the average of all the ratings for that item, and the standard deviation of which is equal to 1.1. Then, the attackers randomly assign a value from that normal distribution to the item as their provided rate for that item. The reason behind this act is that in this way they would resemble the other ordinary users more. On the other hand, for the items about which the recommender system attackers aim to misinform the system, they assign the lowest or highest possible ratings according to their intent.

The above concept can be utilized in review spam detection by considering that behavior from a shilling attacker as a similar behavior from a review spammer. It means that one of the spamming approaches can just be the same. The spammers rate each of the products which are not their spamming targets by randomly selecting a value from a normal distribution corresponding to each product where the mean of the normal distribution is the average of the ratings for each product, and the standard deviation is 1.1. The incentive for spammers to behave like this with the products they do not intend to spam can be explained as follows. By giving ratings close to the average ratings the spammers would look like an average reviewer and so they will not seem suspicious. Again similar to the shilling attackers, the spammers give the highest or the lowest ratings to their targeted products regarding their purpose.

In the proposed approach of this section, the concern is to spot the above mentioned type of spammers. This method investigates all the reviews for each reviewer and categorizes his/her reviews into one of the two groups; the reviews on targeted products which include the reviews with extreme ratings, and the reviews on non-targeted products which include the reviews with the ratings which deviates at most by 1.1 from their

corresponding average ratings. For every reviewer, if at least one review is detected which does not lie into any of the above two groups, that reviewer is not likely to be a spammer. Consequently, all the reviewers whose reviews are all placed into one of the above two categories will be announced as the detected potential spammers.

For example, consider the reviewer with the ID of "A2D3JLI2TGK1RV". All of his/her reviews existing in the system for all his/her reviewed products are shown in Table 3-3. The first column shows the IDs of the products, while the second column shows his/her ratings of the products. It can be seen that all the ratings of his/her reviews are whether extreme, i.e., equal to one or five, or within a 1.1 distance from their corresponding average ratings. Therefore, this reviewer is to be announced as a potential spammer.

Table 3-3 A list of the products reviewed by one of the potential spammers along with his given ratings to each product and the average of the ratings for each product

Review	Product ID	Reviewer's Rating	Average Rating
#1	B00000117J	5.0	3.75
#2	B00005N7RF	3.0	2.3333333333333335
#3	B00006KNXP	5.0	2.5
#4	6305873461	5.0	3.3333333333333335
#5	0743457900	5.0	3.5
#6	B000EXZFS0	4.0	3.3333333333333335
#7	0792844068	5.0	2.6666666666666665
#8	B0002JJBZY	4.0	3.875
#9	0471404195	5.0	2.3333333333333335
#10	B0002IQFEA	3.0	3.0

Now, consider another reviewer with the ID of "A2YW7RGRPJEMWR" whose reviews are given in Table 3-4. It can be seen that he/she has published some extreme-rated reviews as well as some reviews with 1.1 deviations from their corresponding average ratings. However, he/she still has some reviews which do not fall into any of the above two categories i.e. reviews number 7, 9, 10, and 14. Therefore, this reviewer is not likely to be a spammer, and so will be regarded as a legitimate reviewer in the system.

Table 3-4 A list of the products reviewed by one of the reviewers along with his given ratings to each product and the average of the ratings for each product

Review	Product ID	Reviewer's Rating	Average Rating
#1	0819844896	5.0	2.5
#2	B000056JLV	5.0	3.0
#3	B00002EIVN	4.0	4.25
#4	B00005BWWT	3.0	3.6
#5	B00000IZQI	5.0	4.315789473684211
#6	B000056HKF	5.0	3.75
#7	B000058DQU	**3.0**	1.5
#8	B000056JLT	5.0	3.75
#9	B000056JAE	**4.0**	2.0
#10	B00005BT6H	**4.0**	2.0
#11	B00005BY8W	3.0	2.3333333333333335
#12	B000056HNQ	2.0	2.3333333333333335
#13	B00005BXJQ	5.0	2.5
#14	B00005BZ71	**4.0**	2.0
#15	B00005MKYT	4.0	3.0
#16	B000056J70	5.0	3.3333333333333335

3.2.3 Spammers with Dense Regions in Their Timeline

It has been argued in this approach that ordinary reviewers usually write their comments on different products in almost consistent patterns during different periods. This expression means that their reviews are distributed almost uniformly over the time frame, from when they have started to write their very first reviews to the present time. Although some time intervals might exist in which they have reviewed several products and so those intervals would seem slightly dense compared to other intervals, generally the normal reviewers' reviews are steadily given over time whilst they experience a new product. It means that they write their reviews only when they have actually utilized the corresponding products, and since individuals can examine a limited number of products at a time, it is unlikely that they will be able to publish many reviews in a short time span. On the other hand, spammers are expected to post their spam reviews on the intended products in a short time span, say in two or three days. Therefore, some relatively small time intervals in which the spammers have published so many spam reviews can be found in the total time of their existence in the system, i.e., from their first presence to the present time. Consequently, if such dense regions are detected in the timeline of a reviewer, that individual will be considered as a suspicious case of being a potential spammer.

As an example consider the reviewer with the ID of "A1000FM37CEEJ9" whose reviews can be found in Table 3-5. It can be obviously observed from the table that 10 out of all his 13 reviews have been published within two days, February 1st and 2nd 2003. It is not practical for a real reviewer to examine ten different products enough to write reviews

about them in only two days. Thus, the discussed reviewer is highly suspicious to be a potential spammer.

Table 3-5 All a reviewer's reviews showed by their reviewed product, posting time, and rate

Review	Product ID	Review Time	Review Rating
#1	0911564020	January 19, 2003	5.0
#2	B00005KCLJ	February 1, 2003	5.0
#3	B000026BEC	February 1, 2003	4.0
#4	B000002SMC	February 1, 2003	4.0
#5	B00000DCI0	February 1, 2003	3.0
#6	B0000037EY	February 2, 2003	5.0
#7	B0000037CH	February 2, 2003	5.0
#8	B000002SUC	February 2, 2003	3.0
#9	B00000AFHA	February 2, 2003	2.0
#10	B00004SDJR	February 2, 2003	4.0
#11	B0000057QR	February 2, 2003	5.0
#12	B00005NG1J	April 6, 2003	3.0
#13	B00004SBWU	April 6, 2003	3.0

3.3 Detecting Spammer Groups

A substantial fact that should be taken into account while studying review spam is the fact that many reviewers work in groups to apply their spamming intentions. There are several reasons for the spammers to form groups rather than acting alone. For example, working in groups will assist them in having huge impacts on the overall opinions about their targeted products. While just one review from one reviewer cannot make a sensible effect, a sufficient number of reviews from several reviewers can readily change the orientation of the opinion about a product. Moreover, reviewers acting as a group are able to compensate each others' suspicious behaviors in order to reduce their probability of getting caught. Thereupon, some approaches to detect spammer groups have been proposed in this section.

3.3.1 Considering Outlier Reviews of Products

In this method, the idea is to detect the spammers by utilizing the outlier reviews of each product. Hence, the first step would be to detect the outlier reviews. An outlier review is a review the rating of which deviates considerably from the average rating of its corresponding product [3]. The proposed algorithm to find outliers for a single product is to calculate the distance between the rating of each review and the average of the ratings of all other reviews for that product. Those reviews the computed distances of which are extremely high compared to other reviews are considered as potential outlier reviews.

As an example consider the published reviews for the product with the ID of "006001315X". All of these reviews can be found in Table 3-6. The computed average rating for the product is 4.1765. Looking at the rightmost column which shows the

distance of the ratings from the average rating, we can mark the 12th review as the outlier review for this product.

Table 3-6 All reviews on a product, and the distance of their ratings from the average rating

Review	Reviewer ID	Review Time	Review Rating	Distance from Average
#1	A19JYLHD94K94D	June 16, 2005	5.0	0.875
#2	A1CDZM5YMB61PD	December 2, 2003	4.0	0.1875
#3	A1I2O9Y3X3HXLS	October 16, 2003	5.0	0.875
#4	A1M4NJYP0WNL8Q	March 6, 2004	5.0	0.875
#5	A1OM1ORZYCZ8VY	October 27, 2003	4.0	0.1875
#6	A1WU1Y1MX9U71V	December 1, 2003	4.0	0.1875
#7	A24MUQNWPDWZIH	May 17, 2004	3.0	1.25
#8	A280Q86OH9DLRZ	November 15, 2003	5.0	0.875
#9	A2CR57GAJKNWVV	October 18, 2003	5.0	0.875
#10	A2KUBN3WS86EW3	July 31, 2004	5.0	0.875
#11	A3DQWFWINN3V5A	October 14, 2003	3.0	1.25
#12	A3E4CX5FKM4ORK	November 11, 2003	1.0	3.375
#13	A3FZ06XRKW5JC5	December 28, 2003	3.0	1.25
#14	A3QVI57VT1VGRO	October 1, 2003	4.0	0.1875
#15	AFVZXHIUSXINA	July 26, 2004	5.0	0.875
#16	ALOESZ0U0FVKZ	September 10, 2004	5.0	0.875
#17	AN22K7319SN21	November 28, 2004	5.0	0.875

Generally, outliers are not necessarily review spam since it might be the case of various tastes or expectations of different reviewers. However, if a group of reviewers, all of whom have reviewed some identical products, are found such that for each of their common products one of those reviewers has an outlier review and the ratings of the

reviews of the rest of the group members are deviating by a slightly small value from the average rating but with the same polarity of the outlier review, that group is very likely to be a spammer group.

For example, assume that a group of five reviewers have reviewed five different products with different ratings. Note that these reviewers might have reviewed other products, and also some other reviewers might have reviewed these products. But, the combination of these five reviewers with these five products satisfies the conditions of the proposed spam detection method.

The reviewers, their targeted products, and their given ratings are illustrated in Table 3-7.

Table 3-7 An example of the ratings of the reviews given by five reviewers on five products

	Reviewer #1	**Reviewer #2**	**Reviewer #3**	**Reviewer #4**	**Reviewer #5**
Product #1	**1.0**	4.0	5.0	3.0	4.0
Product #2	3.0	**5.0**	2.0	2.0	1.0
Product #3	4.0	5.0	**2.0**	5.0	4.0
Product #4	3.0	4.0	4.0	**1.0**	5.0
Product #5	1.0	3.0	1.0	2.0	**4.0**

It can be seen from the information given in the table that Reviewer #1 has an outlier review on Product #1 whereas all the other reviewers have also reviewed that product but with close to mean ratings. So firstly, all the reviewers have commented on all the given products, and secondly, each time one different reviewer has the outlier review which is

the only outlier review for the targeted product. Therefore, this group is highly likely to be a spammer group.

The incentive for this conclusion is that the members of a spammer group do not write several outlier reviews in order not to get discovered easily. In fact, they collaborate together by distributing the act of writing outlier reviews among themselves [11]. So they might write reviews with ratings close to the average but with the same polarity as of the outlier to bias the average towards their intended polarity.

As an example of the real dataset, the members of one of the detected spammer groups are shown in Table 3-8 along with the products for which they have given outlier reviews.

Table 3-8 Spammers of one group and the products on which they have posted outlier reviews

Reviewer ID	**IDs of the products on which they have given outlier reviews**
A10VOEBL5S337W	0064401871, 0064472132, 0064472795, 0345407865, 0312966326, 0425119653, 0439129095, 0439223504, 0590684841, 0671025449, 0671025457, 0689817851, 0689824750, 6304362498
A17U0GUNXP5WD6	0689817851
A1QK1LZMQG5WI7	0689817851, 0684825538, 1573223042
A28OPJYIDHW3OQ	0066214130, 0684801523, 0689817851, 0805211039, 1570625549, B000002MF9, B00008BXJF
A2FBIF1FKBC193	0352332344, 0352333375, 0352334983, 0425173534, 0425198219, 0440224675, 0440226104, 0671742493, 0689817851, 0718003586, 0802136648, 0812590961, 0895260506, B00000K32N, B00001O2GH, B00003CX5P, B00003CXGV, B00003CXRA, B00003CXTA, B00003CXWG, B00003CY5J, B00004SZ6O, B00004Y87P, …

Reviewer ID	IDs of the products on which they have given outlier reviews
A2JQEY8GXSF8IZ	0142002283, 038533379X, 0689817851, 0967686563, 1400031699, 1586481843, 1929125305, B000000WCM, B000001DTM, B0000025E6
A2K4UF4D6YYP14	B0002W4SWC, B000ASDFGI, B000AXSN5G, 0446531138, 0689817851, 0767821769
A33PMNAFRQVP6Q	0689817851, 1888047054
A36L47A45ZF3WP	0689817851, B0000AOX09
A3N9BC4JEFHN3E	B0000000LI, B000002GCP, B000005H7D, B00001OH7T, B00004S95R, B00005ONMT, B00006CY6H, 0689817851
AN5EUMBU37681	0060950668, 0689817851, 0965353362
AR9WK70Z3WI4	0743250605, 0761135901, 0689817851, B00009V3KM

The approaches introduced in this chapter possess different levels of complexity, and so have been suggested to deal with the spammers with different attitudes and various levels of expertise.

All the approaches are explained in details in the next chapter along with their given pseudo codes as well as the spammers detected by each approach.

Chapter Four: Implementation and Experiments of the Approaches towards Review Spam Detection

This chapter explains the detailed steps of the suggested algorithms for the approaches introduced in the previous chapter, as well as their implementations. The methods have been applied to the provided dataset [3], the results are also given for each approach. The order of this chapter is the same as chapter three so that it would be easier to follow each method in its corresponding section. At the end of this chapter, the results of all the proposed approaches have been compared with each other, and the reason why each spammer has appeared in the results of one or more particular methods has been discussed. Afterwards, all the discovered spammers from different approaches were ranked according to the number of their occurrences in the results of each approach. Hence, those spammers who have been detected through all the proposed approaches appear at the top of the list, whereas those who have been detected via only one of the approaches will come at the bottom. Finally, the reviewers with higher ranks in the list would be perceived as the actual spammers more confidently.

4.1 Investigating the Trends of Reviews for the Products

The proposed approaches in this section have investigated the problem of review spam detection by considering the patterns that exist in the set of reviews for each product. These methods have discovered suspicious activities by discovering the unusual cases in the trends of reviews.

4.1.1 Statistics of the Reviews with Positive, Neutral, and Negative Ratings

The idea of this approach is to find the intervals of time in which the common ratios of reviews with positive, neutral, or negative ratings do not hold any more, separately for each product. The suspicious reviewers are then extracted from the detected intervals. To distinguish the abnormal subsets of reviews of different time slots, I decided to sort the reviews of each product according to their posting times. Then, I defined a sliding window of the size of twenty percent of the number of reviews. I repeated the same process with fifteen percent as well as twenty five percent of the reviews. However, the experiments with fifteen and twenty five percent of the reviews were not good enough since the intervals were whether a bit too small or a little too large to accurately simulate the trends, i.e., the sliding window with the size of twenty percent highlights the suspicious intervals, better. The computed ratios with the sliding window of fifteen percent for the product with the ID of "014029628X" have been shown in Table 4-1. Similarly, the computed ratios with the sliding window of twenty five percent for the same product have been given in Table 4-2. Values of both tables can be compared to the values of Table 3-1 which illustrates the ratios for the same product but with the sliding window of the size of the twenty percent of all the reviews.

Table 4-1 Ratios of positive, negative, and neutral reviews with the sliding window of size 15%

Interval	Percentage of Positive Reviews	Percentage of Neutral Reviews	Percentage of Negative Reviews
1st (the earliest reviews)	0.84	0.11	0.03
2nd	0.84	0.11	0.03
3rd	0.84	0.11	0.03
4th	0.84	0.11	0.03
5th	0.84	0.11	0.03
6th	0.8	0.15	0.03
7th	0.8	0.15	0.03
8th	0.8	0.15	0.03
9th	0.8	0.15	0.3
10th	0.8	0.15	0.3
11th	0.84	0.15	0.0
12th	0.84	0.15	0.0
13th	0.84	0.15	0.0
14th	0.84	0.15	0.0
15th	0.84	0.15	0.0
16th	0.84	0.11	0.03
17th	0.84	0.11	0.03
18th	0.84	0.11	0.03
19th	0.84	0.11	0.03
20th	0.84	0.11	0.03
21st	0.8	0.15	0.03
22nd	0.84	0.11	0.03
23rd	0.84	0.11	0.03
24th	0.84	0.11	0.03
25th	0.84	0.11	0.03

Interval	Percentage of Positive Reviews	Percentage of Neutral Reviews	Percentage of Negative Reviews
26th	0.84	0.11	0.03
27th	0.84	0.11	0.03

Table 4-2 Ratios of positive, negative, and neutral reviews with the sliding window of size 25%

Interval	Percentage of Positive Reviews	Percentage of Neutral Reviews	Percentage of Negative Reviews
1st (the earliest reviews)	0.82	0.13	0.04
2nd	0.82	0.13	0.04
3rd	0.82	0.13	0.04
4th	0.82	0.13	0.04
5th	0.82	0.13	0.04
6th	0.78	0.17	0.04
7th	0.78	0.17	0.04
8th	0.82	0.17	0.0
9th	0.82	0.17	0.0
10th	0.82	0.17	0.0
11th	0.82	0.17	0.0
12th	0.82	0.17	0.0
13th	0.86	0.13	0.0
14th	0.86	0.13	0.0
15th	0.86	0.13	0.0
16th	0.82	0.13	0.04
17th	0.82	0.13	0.04
18th	0.82	0.13	0.04
19th	0.86	0.08	0.04
20th	0.86	0.08	0.04

Interval	Percentage of Positive Reviews	Percentage of Neutral Reviews	Percentage of Negative Reviews
21st	0.82	0.13	0.04
22nd	0.82	0.13	0.04
23rd	0.82	0.13	0.04
24th	0.86	0.08	0.04

Comparing Table 3-1 and Table 4-1, we can clearly see fewer fluctuations in the values of Table 4-1, especially in the computed percentages of positive reviews, i.e., the second columns. This means that selecting a sliding window of size 15% of the reviews will not reflect the trend of the ratings of reviews properly since each time 85% of the reviews are involved in the calculations. The existence of most of reviews in the calculations fades out the effects of sudden changes in the trend.

Considering the trend of the ratings of the reviews for the above-mentioned product, i.e., the product with the ID of "014029628X" in Table 4-3, we can see a rating of 1.0 in the middle of the table. This sudden drop in the trend should impact the calculated percentages for the intervals which contain that review. Consequently in the absence of that review, an abrupt increase in the trend of the values calculated as the proportions of positive reviews can be seen in the middle (14^{th} and 15^{th} intervals) and the middle of the lower half (20^{th} review) of Table 3-1 which makes the corresponding reviews get labeled as candidate spam reviews. However, this increase has only occurred in the middle of the lower half (19^{th} and 20^{th} intervals) of Table 4-2 but nowhere in its middle. Moreover, another least frequent value exists in the table in the same column (6^{th} and 7^{th} intervals) which makes it indeterminate to detect the real suspicious interval. Furthermore, no

sudden change has occurred in the corresponding area in the trend in Table 4-3 and thus the sudden decrease in the 6^{th} and 7^{th} rows is unexpected. This is the reason for not preferring the sliding window of size 25% of all reviews.

Table 4-3 All the reviews of the product with the ID of "014029628X"

Review	Reviewer ID	Review Time	Review Rating
#1	A10JPWFOKX42PK	October 13, 2000	4.0
#2	A19N3GRTJ0S8J8	October 24, 2000	4.0
#3	A239W4F69O4MQE	January 14, 2001	4.0
#4	A23GFTVIETX7DS	January 23, 2001	5.0
#5	A295BBQ2M5YM9Q	February 11, 2001	3.0
#6	A2B21POKQ3N09H	March 11, 2001	4.0
#7	A2EGK0YRDF4ZZB	May 19, 2001	5.0
#8	A2MF2QVSCUI27G	June 20, 2001	4.0
#9	A2RZ9O4PSL16V4	July 6, 2001	5.0
#10	A2SHQJP6PNQTLD	August 24, 2001	4.0
#11	A2UDGZUEYHULS5	August 30, 2001	4.0
#12	A2WZQ7TKY0XC5O	September 11, 2001	5.0
#13	A31XWE5EYPB4WW	October 30, 2001	4.0
#14	A36E0YFW6USU8Y	November 8, 2001	5.0
#15	A36USMIZGFO19N	November 13, 2001	**1.0**
#16	A3BIWTN2DA0YY2	January 29, 2002	4.0
#17	A3D7MHV6FTCKW7	February 6, 2002	5.0
#18	A3GT2VY34AXBOJ	February 18, 2002	4.0
#19	A3OW5A5BAKJ25A	April 24, 2002	5.0
#20	A3PQY6QEZ2XU9R	May 21, 2002	3.0
#21	A8OU2FHE8QWFS	May 26, 2002	5.0
#22	AC1K4OQOZ90RS	May 31, 2002	4.0

Review	Reviewer ID	Review Time	Review Rating
#23	AI9GAZ416TIM1	July 13, 2002	4.0
#24	AJ94N3I1A434D	July 22, 2002	4.0
#25	AN0XWUHSHRUG6	December 5, 2002	4.0
#26	AVBTH2XWN74OR	December 23, 2003	3.0
#27	AWG592DW60BKN	January 26, 2004	4.0
#28	AZOGE2ZDSWXV9	April 29, 2004	4.0
#29	A11NGGHK1FU0XV	July 24, 2004	4.0
#30	A1237ROTM7659	January 28, 2005	5.0
#31	A16SLN8OIIK78B	June 27, 2005	3.0

The reason why I have taken an interval of size twenty percent is that my intention was to each time eliminate a small subset of the reviews, and examine the ratios in the absence of that subset. If some noticeable change is revealed in the ratios computed in the absence of the given subset, compared to the ratios computed with the presence of that subset, it means that subset contains suspicious reviews.

Thus, for each product, the first twenty percent of its reviews have been selected as the targeted subset, for the first step. Then, the ratios of positive, neutral, and negative rated reviews for the other eighty percent of reviews have been calculated. The sliding window has been shifted to the second review, and the ratios have been calculated for the new subset of eighty percent of the reviews, as the next step. The same process has been repeated several times, each time by shifting the sliding window one to the right in the timeline of reviews. As a result, all possible subsets have been taken into consideration. To perform the above task over the set of reviews of each product, all the products should have at least five reviews so that for any step, where twenty percent of the reviews should

be withdrawn from the calculations, at least one review can be put away to calculate the ratios for the rest of the reviews. For the products with four reviews or less, the twenty percent of the reviews would be rounded to zero, and so no review would be eliminated from the calculation. As a result, every time the ratios would be calculated for all the reviews and each time the results would be just the same. Hence, the products with less than five reviews have been eliminated from this approach. Figure 4-1 illustrates the corresponding algorithm to compute the ratios. Having been applied on the dataset, the algorithm has provided separate tables of ratios for each of the 44824 eligible products, i.e. those products with no less than five reviews.

Figure 4-1 An algorithm to find the ratios of the positive, negative, and neutral reviews

```
For every product
      For i = 1 to n * 0.8  // n is the number of the reviews of the given product
            pos = 0         // counts the number of the positive reviews for the given product
            neg = 0         // counts the number of the negative reviews for the given product
            neu = 0         // counts the number of the neutral reviews for the given product
            For every review, j, not in the interval of i to i + n * 0.2
                  If the rating of the jth review is positive      pos = pos + 1
                  If the rating of the jth review is negative      neg = neg + 1
                  If the rating of the jth review is neutral       neu = neu + 1
            Calculate and save the percentages of positive, negative, and neutral reviews
```

Having the ratios for all possible subsets of reviews for a product, I have then compared them to detect the most suspicious interval. The idea is to find the least frequent ratio among the ratios of positive-rated as well as negative-rated reviews, for each single product. Basically, there should have been some sudden change in the percentages of

positive and negative ratings in the interval in which the least frequent ratio has been calculated. More clearly, the elimination of a specific twenty percent of reviews in calculating the ratios has resulted in an abrupt change in the values of the ratios. It means that as long as the reviews in that suspicious interval have been utilized in calculating the ratios, their impact on the computed values of ratios was persistent. However, when the sliding window was shifted towards them, the gradual elimination of them in calculations has made the ratios to indicate some changes. Finally, at the time when the sliding window has covered all the reviews in the interval, i.e., while all the reviews in the interval have been eliminated from the calculations, the most deviation from the ratios computed through other windows could be observed. Therefore, the suspicious review or reviews were most probably among that subset of twenty percent of the reviews.

The least frequent ratio might be one of the calculated percentages of the number of positive-rated reviews or negative-rated reviews. If it is the ratio of positive-rated reviews, and if the computed ratio is greater than the average ratio of positive-rated reviews, those reviews with negative ratings in the corresponding time interval are the suspicious reviews since their absence has resulted in a higher value for the ratio. If the least frequent ratio is the ratio of positive-rated reviews but its value is less than the average ratio, the reviews with positive ratings are suspicious. Correspondingly, if the least frequent is the percentage of negative-rated reviews, and its value is greater than its average value, the positive-rated reviews in the interval are suspicious. Finally, if the percentage of the negative-rated reviews is the least frequent, and its value is less than the average value, the reviews with negative ratings are suspicious. The owners of the

detected suspicious reviews from the above four possible cases are then the candidate spammers. The algorithm to detect the candidate spammers is given in Figure 4-2.

Figure 4-2 An algorithm to detect all the suspicious reviewers

```
For every product
      For all the values of its computed positive and negative ratios
            Find the least frequent value, its polarity, and its corresponding interval
      PosAvg = Find the average of the positive ratios computed for all intervals
      NegAvg = Find the average of the negative ratios computed for all intervals
      If (polarity = positive) and (value of the least frequent ratio > PosAvg)
       Or (polarity = negative) and (value of the least frequent ratio < NegAvg)
            For each of the reviews in the detected interval
                  If the review's rating is negative
                  Add its reviewer to the candidate spammers list
      If (polarity = positive) and (value of the least frequent ratio < PosAvg)
       Or (polarity = negative) and (value of the least frequent ratio > NegAvg)
            For each of the reviews in the detected interval
                  If the review's rating is positive
                  Add its reviewer to the candidate spammers list
```

Reviewers, who happen to appear in the list of the candidate spammers several times because of being marked as suspicious reviewers in different intervals of the same product or various products, are to be announced as the found spammers. In my experiment, I have considered those reviewers who have appeared more than ten times in the candidate spammers' list as the detected spammers since an acceptable number of spammers would be detected by considering the total number of the reviewers.

Table 4-4 indicates some of the 14072 detected spammer candidates by their IDs, as well as the frequency of the occurrence of each of them in the list of suspicious reviewers which equals the number of its suspicious reviews.

Table 4-4 IDs of some of the suspicious reviewers and the number of their suspicious reviews

Reviewer ID / Frequency	**Reviewer ID / Frequency**	**Reviewer ID / Frequency**
ASR1YTZS9HDP1, 1	A1G56KHOUOFWDW, 55	A15OVP5X01D4U, 4
A192NLEQY9B5I, 6	A12EC2L7BMJM3R, 8	APUHDIL2CEI3C, 1
A1D9Z20TCUH6ZX,21	A1D7T6QRK3TBLY, 5	A1F7CMDX6QDJVA,15
A1D2C0WDCSHUWZ,	A2EJP1CB7YGPNK, 5	AI5CFOPHQXROO, 5
AR2DE47VCY1C8, 13	ADME58Z2TE50B, 1	A1M566ZCEEA877,6
A1F8U98L5YP7ET, 2	A1KXONFPU2XQ5K, 16	A1R528E072DF8U, 4
A20IIR0422G3A5, 35	A19R2MO4YZ1IYB,12	A2PGNUJV9CCFSL, 1
A2W5KBV5PT4UPM, 2	A3DBNPEIWN3L44, 3	A3UISFQMOSZJMO, 3
A1KYPBC73SIHFH,1	A1JYOJ9VSN7932, 1	A19N3GRTJ0S8J8,12
A1BI8PUEHA5CHW,60	A364X07NE1XD9, 1	ATRVL71MT2YOJ, 3
AUMTUCSE9YY2X, 1	A3DGVB3T5QJNRE, 1	A1FSI59SI6NCX9, 1
A1FCESYTH4Y1O2,5	A1IMBBVGT354MR, 1	A19JEZQIUE00EH, 1
A1GBOCJ943SP8R, 5	A1LKAYASYEDFR2, 2	A1AKQ1YUS4BT82, 3
A2DVMH3P193R5U, 1	A27BY97QQS36V3, 14	A2A4T3NHN1P5ZS, 1
A21QIKFIGT3XRK, 1	A1U5BRBPMTTGWN, 3	AFVQZQ8PW0L, 174
ASHBDNEGHBFGQ, 1	A3W4GJR5CCADBX, 1	A2VZA7NZR75G3T, 2
A3PGGI7A6XCNF1, 19	A6854SLB33FDB, 8	A12TECTYSRJ3UR, 4
A185HGLJQOH1K7, 3	A1QC2BOU3AL7L8, 5	A1TPJMIG83W12L, 9
A1D6EWJWMV2PC2, 2	A2ELYGBTWY38DH, 1	A1NY28967H5TIQ, 23
A1R48YO4NJHPNP, 2	A12DP14GPRZF7E, 9	A16KF1D1T1X5PS, 36

Finally, the reviewers with high frequencies are considered as spammers. A number of the 661 spammers found by this approach are illustrated in Table 4-5.

Table 4-5 Some Spammers' IDs Found by Their False Impact on Ratios of Positive and Negative Ratings

Reviewer ID	Reviewer ID	Reviewer ID
A1G56KHOUOFWDW	A1D2C0WDCSHUWZ	AR2DE47VCY1C8
A20IIR0422G3A5	A27BY97QQS36V3	AFVQZQ8PW0L
A1NY28967H5TIQ	A16KF1D1T1X5PS	A1D9Z20TCUH6ZX
A19R2MO4YZ1IYB	A19N3GRTJ0S8J8	A1BI8PUEHA5CHW
A1RMHZSWZ7ZEQO	A1SAZB83QFR0W2	A16DPRE4OYKXEZ
A2CR57GAJKNWVV	A18FUHNBP90IB4	A19B0GTBYVO2OL
A12XR2KE0J7TUG	A1796BFN7L774T	A1AFXJ8U72MD6L
A1EI65WJC85U68	A1PKJUAQFGNLSX	A2EEUQ81DTY7G3
A1IU7S4HCK1XK0	A20MN959ZZH2DU	A2B21POKQ3N09H
A1845IJB63D5H7	A1RKD1I8MW1LG6	A1ED4H8T6NXF9E
A1KXONFPU2XQ5K	A3PGGI7A6XCNF1	A1F7CMDX6QDJVA
A1TPW86OHXTXFC	A1QJHZUUMKZYZG	A2386P602G9547
A1K1JW1C5CUSUZ	A17XW1EB8DF217	A1ZPZNULFUAU2D
A18M68DE1Y6W51	A1IOJE0W1NXOSE	A15OAXTD8A1FCS
A13E0ARAXI6KJW	A26JGAM6GZMM4V	A15D1DHDS7NQN2
A2375TJ8NZVA4I	A2CVXUY1EYQGGA	A1OSPY4MSOLE5V
A1NEYE93FWXT36	A1HS3BUBNZJJD6	AC1K4OQOZ90RS
A1MYUO5OIDMNMM	A1EMDSTJDUE6B0	A1LVMQ52YODRMO
A14UM7LOF20W6P	A11A8GWG0IXBZH	A1CT7QHFUI5G4Q
A1MQQEM7W77L62	A1R9QOPV6HVEKF	A1V3IO23FZFF9L

4.2 Analyzing Reviewers Behaviors

This section describes the reviewer-oriented approaches which have concentrated on reviewers' different behaviors to detect suspicious acts committed by spammers. These approaches extract reviewers' habits and behavioral patterns while they are posting their comments on various products. Then, they analyze the patterns provided from different aspects, for example once by considering the times of the posts, once by considering the ratings assigned by the reviewers, and so on. For each reviewer, the approaches aim to discover suspicious behaviors which do not follow the usual behavioral patterns. Exploiting the detected abnormal behaviors, we would be able to determine and announce the candidate spammers.

4.2.1 Reviewers with Extreme Review Ratings

This method has been provided to detect those reviewers who have many reviews with the lowest or the highest possible ratings. As discussed in the corresponding section of the previous chapter, the ordinary reviewers do not usually give full positive or full negative ratings to their criticized products. Therefore, those reviewers who have given many such reviews with extreme ratings are suspicious to be spammers. Although this approach of spamming is not usually used by professional spammers because of its simplicity, and the fact that they can easily get caught, there are still some amateurs who utilize this method to abruptly change the overall polarity of the opinions of their targeted products. In my dataset, ratings have been published in the scale of one to five. So, the reviews with ratings equal to one or five were considered as extreme cases, and those reviewers who had many such reviews were introduced as spammers. To distinguish this

kind of spammers, first I have counted the number of all reviews as well as the number of extreme-rated reviews for every reviewer separately. Then, I have selected those reviewers more than %95 of whose reviews were reviews with extreme ratings as the detected spammers. The corresponding algorithm is given in Figure 4-3.

Figure 4-3 An algorithm to find the spammers who have many reviews with extreme ratings

```
ArrayList<ArrayList<String>> reviewers = new ArrayList<ArrayList<String>>()
                //Each row in the reviewers arraylist saves a reviewer-ID, his total
                number of reviews, and the number of his extreme-rated reviews
For i = 1 to n  // n is the number of all the reviews exist in the dataset
     boolean flag = Find the reviewer-ID of the ith review in reviewers
     if flag = true
          Add one to the number of his reviews
          If (the ith review's rating = 1.0) or (the ith review's rating = 5.0)
               Add one to the number of his extreme-rated reviews
     if flag = false
          Add him to reviewers and set the number of his reviews to 1
          if (the ith review's rating = 1.0) or (the ith review's rating = 5.0)
               Set the number of his extreme-rated reviews to 1
          else
               Set the number of his extreme-rated reviews to 0
For every reviewer in reviewers
     if (the number of his extreme-rated reviews / the total number of his reviews) > 0.95
          Mark him as a spammer
```

I have done the same task two other times; once I selected the reviewers more than 80% of whose reviews were extreme-rated, and another time I chose the reviewers more than 90% of whose reviews were extreme-rated. The former resulted in detecting 6120 spammers, and the latter resulted in detecting 3012 spammers. Comparing the number of

spammers found by utilizing the approach proposed in this section, while taking reviewers with more than 80% and 90% extreme-rated reviews, with the results provided by the other proposed methods; I recognized that too many reviewers are announced as spammers. However, the number of detected spammers 95% of whose reviews are extreme-rated seemed to be more acceptable since it is close to the numbers of spammers detected by other approaches. Consequently, I have reported 1928 reviewers as spammers some of whom can be found in Table 4-6.

Table 4-6 A Few Spammers' IDs Found By Their Significant Number of Extreme-rated Reviews

Reviewer ID	**Reviewer ID**	**Reviewer ID**
A104VHIMHTA3V9	A1075NTSI7NRLJ	A1087DECRN5UDU
A10A0MBFUQ3V81	A10AECKZ0GRKG	A10DQ97EP60BVG
A10ZG0QKITDBW1	A10ZSSZO3BO11Z	A111AUAVWX6YIQ
A118GK08650JY7	A1192ZVWRV7HDO	A11C3BT4A8M9VG
A11CH0J2O9MR1Y	A11CNB9UQ0VVP	A11GXFDML3YAOU
A11J6S7N0LIPOY	A11PXVDPJDIZY8	A11V8J25BDD0AH
A123A9IBBMYQV9	A123QQMMA4GRF5	A129CGGSGVOA0Q
A12E1N994TQIH7	A12HCSRSP7NHT1	A12JAOY8Q0SLW0
A12KVAXA4T21QL	A12QQXGSSHWQ21	A12VLMC3AVOV0J
A12YJBN58HJIG	A134BCJKZ0Z043	A137GQ32Q97F6C
A139BEPV96YOXG	A13CADRZQEEZ23	A13CTR8SOSER5V
A13F94JCFUX9RY	A13MWIWORVRQ5S	A13NCBZ02TPRLN
A13RVB474Y25MW	A13V1OMU4SP9H7	A13WLUIATM3NW2
A13Y57V8MRPUZN	A142UV53K381N9	A144R5DP2PMFXI
A146FX9WCSLVXL	A146QCGOQPZLTG	A147SS0ZPZPPFC
A148UWKWUZ8FON	A14FIKDYV9NMZP	A14JZ8IGLJFOO4
A14LIZKKA7WYT9	A14PPHRETAWA0C	A14VYQT8DPBI9F

Reviewer ID	Reviewer ID	Reviewer ID
A14W0N8PWQAKYU	A14YXWOFJ9LTSK	A153QAYQJVVM0H
A156QNPXZ736ZB	A15A7XGROJQBT9	A15B5DO3A3A30C
A15IO2K28WY330	A15JE4OCMMASMH	A15LH1T332KVNR

4.2.2 Spammers with Close to Mean Ratings for Non-targeted Products

The purpose of this approach is to find those reviewers who rate some products, which they aim to praise or belittle, with the highest or lowest possible values while rate the rest of the products with ratings close to the mean of the ratings of the products, in order to resemble the majority of reviewers and so not to seem suspicious. To detect such spammers, first I have calculated the average ratings of the reviews for each of the 474,472 products, separately. Some of the results can be found in Table 4-7.

Table 4-7 Some of the products representing by their IDs with their calculated average ratings

Product ID	B0007RT9LC	B00028HBKM	B00062IVM6
Average Rating	4.142857142857143	3.6294416243654823	4.153846153846154
Product ID	B00064LJVE	B0002GMSC0	B00019RD1Y
Average Rating	3.1510791366906474	2.5289256198347108	2.5
Product ID	B00003CXHM	B000059H9C	B00000153R
Average Rating	3.9411764705882355	3.4	2.5
Product ID	B00003CXE7	0691008752	0783228473
Average Rating	4.10091743119266	2.0	4.0

Then, I have investigated for all the reviewers whether each of their reviews was an extreme-rated review, i.e., with one or five rating, or was deviating from the mean of the ratings of its corresponding product by at most 1.1. The reason why I chose the same value of 1.1 for the standard deviation in my problem is the fact that the ratings of the reviews range from 1.0 to 5.0 in my dataset. Therefore, values which are at most 1.1 less or 1.1 more than a specific value in this scale can be regarded as close values to that specific value. For example, if we assume that a rating of 3.0 denotes the neutral opinion of its owner about some product, all the reviews with the rating values from 1.9 to 4.1 can be still considered as neutral opinions.

If a reviewer has been discovered such that at least one of his/her reviews was not satisfying any of the above conditions, i.e., the reviewer has at least one review which is not rated with an extreme value nor with a value close to the average rating, that reviewer would be eliminated from the list of candidate spammers. Hence, in the introduced algorithm I distinguished the normal reviewers first. Then I eliminated them from the list of all reviewers to obtain the final list of the discovered spammers. The proposed algorithm is illustrated in Figure 4-4.

Figure 4-4 An algorithm to find spammers who give extreme ratings to their targeted products and close to mean ratings to their non-targeted products

```
ArrayList<String> reviewers = new ArrayList<String>()
ArrayList<String> innocent reviewers = new ArrayList<String>()
ArrayList<String> spammers = new ArrayList<String>()
For every product
      Find the average of its reviews' ratings
Add the IDs of all the reviewers to reviewers
For i = 1 to n  // n is the number of all the reviews in the system
      if (review(i).rating >= 2.0) and (review(i).rating <= 4.0)
      and ((review(i).rating - its product average rating)² >= 1.1)
            Add review(i).reviewer to innocent reviewers
For every reviewer in reviewers
      if it is not in innocent reviewers
            Add it to spammers
```

After omitting the normal reviewers, the remaining 1655 reviewers were considered as the detected spammers of this type. A few examples of these spammers are illustrated in Table 4-8.

Table 4-8 A few Samples of the Spammers All Whose Reviews Are Extreme-rated or Close to Mean

Reviewer ID	Reviewer ID	Reviewer ID
A104VHIMHTA3V9	A1075NTSI7NRLJ	A1087DECRN5UDU
A10AECKZ0GRKG	A10DQ97EP60BVG	A10ZG0QKITDBW1
A118GK08650JY7	A11CH0J2O9MR1Y	A11CNB9UQ0VVP
A11J6S7N0LIPOY	A11PXVDPJDIZY8	A11V8J25BDD0AH
A129CGGSGVOA0Q	A12E1N994TQIH7	A12HCSRSP7NHT1

Reviewer ID	Reviewer ID	Reviewer ID
A12QQXGSSHWQ21	A12VLMC3AVOV0J	A12YJBN58HJIG
A137GQ32Q97F6C	A13CADRZQEEZ23	A13CTR8SOSER5V
A13MWIWORVRQ5S	A13NCBZ02TPRLN	A13RVB474Y25MW
A13WLUIATM3NW2	A13Y57V8MRPUZN	A144R5DP2PMFXI
A146QCGOQPZLTG	A147SS0ZPZPPFC	A14FIKDYV9NMZP
A10A0MBFUQ3V81	A111AUAVWX6YIQ	A11GXFDML3YAOU
A123A9IBBMYQV9	A12JAOY8Q0SLW0	A134BCJKZ0Z043
A13F94JCFUX9RY	A13V1OMU4SP9H7	A146FX9WCSLVXL
A14JZ8IGLJFOO4	A14PPHRETAWA0C	A14VYQT8DPBI9F
A14W0N8PWQAKYU	A14YXWOFJ9LTSK	A153QAYQJVVM0H
A156QNPXZ736ZB	A15A7XGROJQBT9	A15B5DO3A3A30C
A15IO2K28WY330	A15LH1T332KVNR	A15LVJNNKFPR6U
A15TIQSRJFRW1W	A163FF0PYSTJ5U	A164FSMF5V673Q
A165YQNR22LWEJ	A167ABU99FT4OC	A16L6QCBBYOUZX
A16NK4WD84MNK8	A16QM98NAYJ64	A16WM7LFPRHZ65

4.2.3 Spammers with Dense Regions in Their Timeline

One of the criteria based on which one type of the spammers can be distinguished from the rest of the reviewers is the existence of the reviews posted by those spammers on the products in a short time span. This means if we map each spammer's reviews to the time axis, some fairly small intervals can be found in which many reviews have been published. As a result, our proposed approach has endeavored to detect the spammers by investigating reviewers' timelines to find dense regions in which the reviewers have published a considerable number of reviews. To implement this algorithm, first I needed to determine the appropriate time interval to be considered as a short interval for my

dataset. Therefore, I took a sliding window with the size of three days. This is because after investigating different individuals' reviews, I have found that most of the reviewers have usually posted all of their reviews in an interval of two months. The suitable short time interval can also be found by trial and error. First, an arbitrary number of days can be selected as the short interval, and the algorithm would be operated. If the number of the detected spammers forms a considerable percentage of the number of all reviewers, the selected time interval should become smaller. Conversely, if the percentage is too little, the time interval should become larger. This process must be repeated till a good number of spammers are discovered.

After finding the appropriate short time interval, for each reviewer I have started from his/her earliest review, counting how many reviews he/she has written in three days from the date of that review. If the number of the counted reviews was greater than half the number of all the reviews from that reviewer, he/she would be regarded as a spammer. If the number was less than that, the sliding window was shifted to his/her second earliest review, and the same steps were repeated. This process was continued until when more than half of the reviews which have been posted within three days were found, or when the sliding window has passed half of the reviews. Therefore, for every reviewer, if half of his reviews have been posted in a time interval of three days, he would be considered as a suspicious spammer. A pseudo code of the given algorithm is given in Figure 4-5.

Figure 4-5 An algorithm for detecting the spammers who have published more than half of their reviews in three days or less

```
ArrayList<String> SuspiciousReviewers = new ArrayList<String>()
For i = 1 to n  // n is the number of reviewers
      Read all the reviews from the ith reviewer
      For j = 1 to m / 2    // m is the number of the reviews from the ith reviewer
            int count = 1
            Date = posting date of the jth review
            For k = j + 1 to m
                  If |posting date of the kth review – Date| < 3 days
                        count++
                  else
                        BREAK
            If count > m / 2
                  Add the ith reviewer to SuspiciousReviewers
                  BREAK
Print SuspiciousReviewers into file
```

Applying the above explained algorithm to all the reviewers, I have discovered 2134 spammers a few of whom are shown in Table 4-9.

Table 4-9 A few Samples of Spammers Detected from Dense Regions of Reviews in Their Timelines

Reviewer ID	Reviewer ID	Reviewer ID
A1000FM37CEEJ9	A100MU000SQ0QB	A100TFWISFG91K
A1021X1F41GBCJ	A103FSOEF5ELTU	A103M039GIFX02
A104J1AT0BH6E2	A10178O9EFHU4E	A106GGUYFEK2NW
A10795OBCFM52U	A1072FAO6ZNJD1	A1087DECRN5UDU
A108XII8MV9XRD	A10AZ52KX1UM1N	A10A0MBFUQ3V81
A10B8DKR3OYZYT	A10DMZD2HDLHUZ	A10DQ97EP60BVG
A10ETSMF2H8QG2	A10F7CLFK5QF5G	A10F8ANXHL4X9Q
A10H3VDEPW7QWI	A10FND101CWIHU	A10KUXWFAWLLPF
A10NG9VLEPYQAC	A10OHNABR03W1V	A10MG7WG0NRHIM
A10UNRYER7C28F	A10VIQSD1GARWC	A10UHOIJ4R52CG
A10VS0ULQ9NU4A	A10W5NFZ9PLX4K	A10X56XICXHDWY
A10XR847DGAC6	A10XZH78SDOI5M	A10Z8FC0SMU5VQ
A10ZZI2LLDEIZX	A110RAQEJP9BYH	A1110WNN5AD4G3
A112EGLR4PM21M	A115CMZF52JA81	A117ZEQ3R1OE0J
A119QEO7ELXJ3W	A11A9HUXY7ZGC0	A11AU06UJMVNTD
A11BHLDCZ6YUQC	A11BSXV5YMAF3K	A11BTXUA99541Q
A11BZL4EPDET73	A11CGGFYC1WC9Y	A11CH0J2O9MR1Y
A11DHNURZ0JJCD	A11DRRDZJXEMZQ	A11GEWV3H9NEV4
A11GZ644T8R4PK	A11H2YWJ0RCMPX	A11HQGBRM8SGUG
A11J6PLQ6DPDIO	A11J6S7N0LIPOY	A11MIUF8419YO6

4.3 Detecting Spammer Groups

Some approaches towards finding the spammers who tend to work in groups rather than working by themselves have been introduced in this section.

4.3.1 Considering Outlier Reviews of Products

For this approach, first I needed to find the distance between the rating of each review and the average of the ratings of other reviews to find the outlier reviews for every product. It means that each time I have excluded the review, for which I was finding the distance, from calculating the average in order to eliminate its impact on the computed average. Therefore, the investigated products should not have less than three reviews so the average can be calculated from at least two values. Thus, I eliminated non-eligible products in this method, i.e., products with one or two reviews. Then, I found the set of outlier reviews for each product. I computed the above mentioned distance for all reviews of a product, and selected those reviews with the distance greater than the average of the minimum and the maximum distance as outlier reviews. I repeated this task for all products. A pseudo code of the above explained algorithm is given in Figure 4-6.

Figure 4-6 An algorithm for finding the owners of the outlier reviews for each of the products

```
ArrayList<ArrayList<String>> Outliers = new ArrayList<ArrayList<String>>()
For i = 1 to n    // n is the number of products
      ArrayList<String> Outlier = new ArrayList<String>()   // It saves ProductID, its average rating, its
                                                            outlier reviewers, and their given ratings.
      ArrayList<String[]> Ratings = new ArrayList<String[]>()     // For each reviewer, it saves his ID,
                                                                  his given rating, and the distance of his
                                                                  given rating to the average rating.
      Add the ith product's ID to Outlier
      For each review of the ith product
            Add the reviewerID and his rating to Ratings
      Find the average rating for the ith product using Ratings
      Add the average rating to Outlier
      For each rating in Ratings
            Find the average of other ratings in Ratings
            Add distance = |this rating - average of other ratings| to Ratings
      min = Find the minimum distance
      max = Find the maximum distance
      For every reviewer in Ratings
            If distance > (min + max) / 2
                  Add the reviewerID and his rating to Outlier
      Add Outlier to Outliers
```

Afterwards for each selected product, I took the owner of one of its outlier reviews and all other reviews with the same polarity of deviation as a candidate spammer group. I formed all the candidate spammer groups by doing so for every outlier review of every product. The algorithm can be found in Figure 4-7.

Figure 4-7 An algorithm for forming spammer groups

```
ArrayList<ArrayList<String>> Groups = new ArrayList<ArrayList<String>>()
For every reviewer in Outliers
      ArrayList<String> Group = new ArrayList<String>()
      Add the reviewerID to Group
      If the polarity of his given rating is negative
            For every other reviewer who have reviewed the same product
                  If their given ratings are negative
                        Add their IDs to Group
      If the polarity of his given rating is positive
            For every other reviewer who have reviewed the same product
                  If their given ratings are positive
                        Add their IDs to Group
      Add Group to Groups
```

Then, I investigated for each group member whether that person had an outlier review for any product where all the other members have reviewed the same product but not with outlier reviews. If the answer was yes, that member would have remained in the group. He would be put aside from the group otherwise. After performing the above task for all the group members, the resulting group can be considered as a spammer group. By repeating the same steps for all the candidate groups, all such spammer groups were detected. The corresponding algorithm is given in Figure 4-8. The first part of the algorithm makes sure each time one of the reviewers has posted the outlier review, whereas the second part investigates if all of the group members have reviewed the targeted product, i.e., if they form a clique.

Figure 4-8 An algorithm to check the group members have reviewed same products while each time one of them has given the outlier review

```
For i = 1 to n  // n is the number of the groups in Groups
      For every reviewer in ith group
            Find the group in which this reviewer has an outlier review
            Check if the owner of the outlier review in ith group exists in this group
            If it does not exist
                  Remove the reviewer from its group in Groups
For each of the groups in Groups
     Sort its members using their IDs alphabetically
For each of the groups in Groups
     int count = Count how many times it has been repeated in Groups
     Keep only one copy of each group
For each of the groups in Groups
     If the number of its members is equal to its computed count
          Save the group as a complete clique   // If the two values are equal, it means that the
                                                group has been repeated by the number of its
                                                members since each time one distinguished member
                                                has published the outlier review.
     else
          Remove the group from Groups
```

As some reviewers have been eliminated from the groups during the previous process, some groups have been retained with only one member. So, I have deleted those groups since they cannot be considered as a spammer group anymore.

In my dataset the number of reviewers was much less than the number of products, and consequently many reviewers have commented on the same products. As a result, 15269 groups were found at this stage. So at the next step, I detected those groups a superset of

which existed among the spammer groups, and eliminated them as well. 15027 groups remained. The corresponding algorithm is given in Figure 4-9.

Figure 4-9 An algorithm for finding maximal groups

```
For i = 1 to n  // n is the number of the groups in Groups
      Compare the members of the ith group with the members of all other groups in Groups
      If some group is found which has at least all the members of the ith group
            Remove the ith group from Groups
```

Finally, according to this intuition that if a candidate group is larger it is more likely that the group is a spammer group since it is less probable that the group is formed by chance [5], I sorted the groups based on the number of their members. The ten largest groups are illustrated in Table 4-10.

Table 4-10 The Spammer Groups Found by Considering Their Outlier Reviews

Group	Reviewers who are the group members
Group #1	A13E0ARAXI6KJW,A15T4TWV79KGQI,A18YMFFJW974QS,A1E2EI93MJW6QK,A1Y01961GKDLO,A1ZDWD35AZL3S,A24A0GJAD2329K,A257IJ95BN71SJ,A27PKWWPO2TXNN,A2M3H5568FCKNZ,A2MJ34EBDEFBNE,A2UIXU97JYCPZG,A2VD2D1QHCMF9Q,A332DUSZF5XC18,A3J3AKELO8I207,A3K891X96IQ2D2,A3VDSGNIS92OVZ,AKTIEXCC0C1AN,AQDJTEIDUKK5B
Group #2	A12RP2HVR5WERP,A15XY6GGKD3B6D,A170QV8HVJ3CMT,A22CN6T98WY8ZZ,A2GDIWLR2VP389,A2KP92WSK7BPIM,A2QPAS1OMRGQPO,A3CTNXKNSNVIYS,A3GA09FYFKL4EY,AAACTCCF2HMQP,AC9B29BXXWJHD,AG2IEP1MJQHFS,AKHJVOYEX6QME,AQSU58L0QIE2G
Group #3	A13E0ARAXI6KJW,A14PTJ17T1GTCE,A1CZICCYP2M5PX,A204906HLJ443E,A26JGAM6GZMM4V,A2GYX971VETQBV,A35UNON2GI8C0R,A3SJSNS9LW981O,A3U5JCXIXTGSLN,A96K1ZGW56S2I,AAI57M3OXP5NK,AOPPHDPD830VJ,AX19A39YU2Y71,AZWX8YF8K2DB
Group #4	A19489DXZMWIR6,A1B1165CZMIT33,A1KYJ99J8WRLF8,A1MI6U0ZH60D36,A1WPZOOHTUBH71,A2B21POKQ3N09H,A2M317B8OJS1YY,A2ZM4AMV7W65V,A37GGSMH6KLS5N,A3BC75ITUEVC93,

Group	Reviewers who are the group members
	A3H99PYJRDSXW3,A3U7MNFK7HKXFE,AJVKTTDLSN7N5
Group #5	A15Z6660PNYIP5,A16SS8HYJW7IEJ,A1JYQVP71WHBTL,A2J57EQ9KPGXR9,A2P0Y85C896QIB,A31U2QT7SAL7K,A366JNHOWLQWQQ,A3OGFKAFLLWK4R,ACFP0K0DNARQM,AM836D28P0EEU,AQP1VPK16SVWM,AT2O4ZW390JCV,AUTBHG6070SL4
Group #6	A170X28DQS49RX,A174UOE2FZU7HW,A1KAUV9QYLP52J,A1T0Z4J5PPLTC7,A29Y34ALWJUGB5,A2CLUVY0XTPRZK,A2JOCPR4EPYOP0,A2Y1D9287MGNQ4,A2ZLROGIL2V7GV,A3989AODHMQ74G,A3Q4BPS221HV1M,A423XLYS1Q1D6,APP45XD85ARYL
Group #7	A1HS9F7J51E9CB,A1NC9AGZOBI0M1,A1RAUVCWYHTQI4,A2T2Z09SPQ0KKZ,A35ESIMG9JLFWB,A3F3UARRAYEQH6,A3H5VO429A9818,A3NCKDPCAUOD4T,A5HMT6ZOBUAVM,A9LAPV8XNKZVZ,AAAS16VM0CST9,AFITDZEWKJKAM
Group #8	A10VOEBL5S337W,A17U0GUNXP5WD6,A1QK1LZMQG5WI7,A28OPJYIDHW3OQ,A2FBIF1FKBC193,A2JQEY8GXSF8IZ,A2K4UF4D6YYP14,A33PMNAFRQVP6Q,A36L47A45ZF3WP,A3N9BC4JEFHN3E,AN5EUMBU37681,AR9WK70Z3WI4
Group #9	A14SE1GA8X31DC,A1GDM2HIU6M21Z,A1JQMPDY8BCFWZ,A291YTUZVJ7G9K,A2C5P3398RHV8I,A2R8TI6YN6NY9C,A2TQURQT6PRRSQ,A37GGSMH6KLS5N,A3GGMPRYCQ87CY,AD0J5KK4WQXNS,AIWSHK5LDNG6Z,AUKLLB4J4YHNA
Group #10	A13E0ARAXI6KJW,A19ZXK9HHVRV1X,A1AG4LMWW9WHDL,A1KAR9G699RM56,A1YQ6QB2127AJ4,A2HG5MY4YOY7N2,A2NJO6YE954DBH,A2T049UQONS0OY,A6ADO7B6FUVN,ACEA95FQS1AVP,AMLV3B3JDF4HD,ATE5SA1VTJNP8

4.4 Final Discussions and Comparisons

In this section, first the proposed approaches are compared with each other so that their common aspects get extracted, and furthermore this will allow me to determine how the approaches can complement each other to give more robust results in review spam detection. Then, the detected spammers are examined to find out in how many approaches they got detected. Finally, the spam reviews are ranked according to the

number of the approaches which have detected them. The spammers who exist in the results of all the approaches are the most likely spammers.

The first approach detects the reviews which have made sudden changes in the ratios of positive and negative ratings. Those reviews are highly probable to be extreme-rated reviews so that they could have effects on the ratios. However, this approach doesn't consider all the extreme-rated reviews. If a reviewer has many, in our experiment 10, of this kind of reviews, he is a spammer.

The second approach considers all the extreme-rated reviews. So, if a reviewer has many, in our experiment 95%, of such reviews, he is a spammer. Although both of these approaches are dependent on the extreme-rated reviews, they employ them from different perspectives. Consequently, the results of none of them are a subset of those of the other one. A spammer might have impact on the ratios of positive and/or negative reviews several times while less than 95% of his reviews are extreme-rated. On the other hand, more than 95% of a spammer's reviews might be extreme-rated but those reviews would rarely have influence on the changes in the values of the ratios.

The third approach detects those spammers all of whose reviews are either extreme-rated or are rated with ratings close to the average ratings of their targeted products. If the extreme-rated reviews of these spammers had impacts on the ratios of positive and/or negative reviews for more than 10 products, they would also get detected in the first approach. In the same way, if these spammers' extreme-rated reviews compose more than 95% of their reviews, they have got caught in the second approach as well. However, the results of this approach are neither a subset nor a superset of the first and the second approaches, necessarily.

The fourth approach considers the posting time of the reviews. Regarding the properties of the dataset, this approach declares those reviewers more than half of whose reviews are given within 3 days as the spammers. The fundamental idea behind this approach is completely different from all the above approaches. Hence, the way the spammers are detected by this approach is totally independent of that of the other approaches. As a result, they might have some common results as well as some particular ones whereas there is no connection between their commonalities.

The fifth approach targets the reviewers who form groups by reviewing the exact same products. Those groups, who have one of their members as the owner of the outlier review for each product while the rest of the members have reviews with close to average ratings, are considered as spammer groups through this approach. Outlier reviews are those which deviate significantly from the average ratings of their corresponding products and therefore not all but many of them are highly likely to be extreme-rated. Consequently, they might be detected by the previous three approaches if they have satisfied their particular conditions. But again, although this approach and the above three approaches may have common results, there still exist several spammers who are discovered by only one approach.

As explained above, each of the proposed approaches investigates the problem of review spam detection from a different point of view, and so each can detect spammers with specific characteristics. Therefore, utilizing them together can provide us with robust results since it combines the examined characteristics all together. The spammers who got caught in most of the introduced approaches are extremely probable to be spammers since they retain many of the spamming characteristics.

A number of the common detected spammers between each pair of the approaches are given in Table 4-11 to Table 4-20.

These reviewers are more likely to be spammers since they have two types of spamming behaviors.

Table 4-11 The spammers detected by both the first and the second approaches

Spammer ID	Spammer ID	Spammer ID
A1M8PP7MLHNBQB	A1XN6MBYDBTX16	A20DZX38KRBIT8
A1B9ISLYYCHIVE	A2OLJM7IREKPWZ	A1G5Q9HBN0EGDV
A1VC6EX5H0IINI	A288SXRFQA42XE	A1FZA9C5MVIWHE
A2U49LUUY4IKQQ	A1JDE9YIFLBR4P	A1XNPOQDLLJJU3
A10ZSSZO3BO11Z	A1DBQURAZBR8T	

Table 4-12 The spammers detected by both the first and the third approaches

Spammer ID	Spammer ID	Spammer ID
A1B9ISLYYCHIVE	A2OLJM7IREKPWZ	A1G5Q9HBN0EGDV
A1DBQURAZBR8T		

Table 4-13 The spammers detected by both the first and the forth approaches

Spammer ID	Spammer ID	Spammer ID
A10B8DKR3OYZYT	A15T4TWV79KGQI	A16EAJTQ59LK8Z
A1A0S7SOGWO9SU	A10W5NFZ9PLX4K	A14TAVG028YD6M
A18HCRSSHL1SML		

Table 4-14 Some of the 647 spammers detected by both the first and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A1G56KHOUOFWDW	A1D2C0WDCSHUWZ	AR2DE47VCY1C8
A1KXONFPU2XQ5K	A20IIR0422G3A5	A27BY97QQS36V3
AFVQZQ8PW0L	A3PGGI7A6XCNF1	A1NY28967H5TIQ
A16KF1D1T1X5PS	A1D9Z20TCUH6ZX	A1F7CMDX6QDJVA
A19R2MO4YZ1IYB	A19N3GRTJ0S8J8	A1BI8PUEHA5CHW

Table 4-15 Some of the 1655 spammers detected by both the second and the third approaches

Spammer ID	Spammer ID	Spammer ID
A104VHIMHTA3V9	A1075NTSI7NRLJ	A1087DECRN5UDU
A10A0MBFUQ3V81	A10AECKZ0GRKG	A10DQ97EP60BVG
A10ZG0QKITDBW1	A111AUAVWX6YIQ	A118GK08650JY7
A11CH0J2O9MR1Y	A11CNB9UQ0VVP	A11GXFDML3YAOU
A11J6S7N0LIPOY	A11PXVDPJDIZY8	A11V8J25BDD0AH
A123A9IBBMYQV9	A129CGGSGVOA0Q	A12E1N994TQIH7

Table 4-16 Some of the 395 spammers detected by both the second and the forth approaches

Spammer ID	Spammer ID	Spammer ID
A1087DECRN5UDU	A10A0MBFUQ3V81	A10DQ97EP60BVG
A11CH0J2O9MR1Y	A11J6S7N0LIPOY	A11PXVDPJDIZY8
A12E1N994TQIH7	A12QQXGSSHWQ21	A137GQ32Q97F6C
A13CTR8SOSER5V	A13WLUIATM3NW2	A156QNPXZ736ZB
A16QM98NAYJ64	A16WM7LFPRHZ65	A16YIC86R49CQK
A16Z7G9BBYH9EN	A16ZPLT8MEVRKA	A17AY5OU3SOE3D

Table 4-17 Some of the 541 spammers detected by both the second and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A10ZG0QKITDBW1	A11C3BT4A8M9VG	A123A9IBBMYQV9
A123QQMMA4GRF5	A12KVAXA4T21QL	A12YJBN58HJIG
A139BEPV96YOXG	A13CADRZQEEZ23	A13CTR8SOSER5V
A13MWIWORVRQ5S	A13RVB474Y25MW	A13V1OMU4SP9H7
A13WLUIATM3NW2	A13Y57V8MRPUZN	A146QCGOQPZLTG
A14PPHRETAWA0C	A15A7XGROJQBT9	A15B5DO3A3A30C

Table 4-18 Some of the 378 spammers detected by both the third and the forth approaches

Spammer ID	Spammer ID	Spammer ID
A1087DECRN5UDU	A10A0MBFUQ3V81	A10DQ97EP60BVG
A10FND101CWIHU	A10MG7WG0NRHIM	A10ZZI2LLDEIZX
A117ZEQ3R1OE0J	A11CH0J2O9MR1Y	A11GEWV3H9NEV4
A11GZ644T8R4PK	A11J6PLQ6DPDIO	A11J6S7N0LIPOY
A11PXVDPJDIZY8	A12260EDCTLAB	A123AY5HLVC2TE
A128JGKDEE11E6	A12E1N994TQIH7	A12HH6HLDPSG19

Table 4-19 Some of the 377 spammers detected by both the third and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A10ZG0QKITDBW1	A123A9IBBMYQV9	A12YJBN58HJIG
A13CADRZQEEZ23	A13CTR8SOSER5V	A13MWIWORVRQ5S
A13RVB474Y25MW	A13V1OMU4SP9H7	A13WLUIATM3NW2
A13Y57V8MRPUZN	A146QCGOQPZLTG	A14PPHRETAWA0C
A15A7XGROJQBT9	A15B5DO3A3A30C	A15IO2K28WY330
A15LH1T332KVNR	A15LVJNNKFPR6U	A15TIQSRJFRW1W

Table 4-20 Some of the 719 spammers detected by both the forth and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A1021X1F41GBCJ	A103M039GIFX02	A10795OBCFM52U
A108XII8MV9XRD	A10AZ52KX1UM1N	A10B8DKR3OYZYT
A10H3VDEPW7QWI	A10NG9VLEPYQAC	A10VIQSD1GARWC
A10W5NFZ9PLX4K	A10XZH78SDOI5M	A112EGLR4PM21M
A115CMZF52JA81	A117ZEQ3R1OE0J	A11A9HUXY7ZGC0
A11AU06UJMVNTD	A11BZL4EPDET73	A11CGGFYC1WC9Y

Tables 4-21 to 4-28 illustrate some examples of the spammers detected by three of the approaches. These reviewers are even more likely to be spammers than the previous ones.

Table 4-21 The spammers detected by the first, the second, and the third approaches

Spammer ID	Spammer ID	Spammer ID
A1B9ISLYYCHIVE	A2OLJM7IREKPWZ	A1G5Q9HBN0EGDV
A1DBQURAZBR8T		

Table 4-22 The spammers detected by the first, the second, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A1M8PP7MLHNBQB	A1XN6MBYDBTX16	A20DZX38KRBIT8
A1B9ISLYYCHIVE	A2OLJM7IREKPWZ	A1G5Q9HBN0EGDV
A1VC6EX5H0IINI	A288SXRFQA42XE	A1FZA9C5MVIWHE
A2U49LUUY4IKQQ	A1JDE9YIFLBR4P	A1XNPOQDLLJJU3
A1DBQURAZBR8T		

Table 4-23 The spammers detected by the first, the third, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A1B9ISLYYCHIVE	A2OLJM7IREKPWZ	A1G5Q9HBN0EGDV
A1DBQURAZBR8T		

Table 4-24 The spammers detected by the first, the forth, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A10B8DKR3OYZYT	A15T4TWV79KGQI	A16EAJTQ59LK8Z
A1A0S7SOGWO9SU	A10W5NFZ9PLX4K	A18HCRSSHL1SML

Table 4-25 Some of the 378 spammers detected by the second, the third, and the forth approaches

Spammer ID	Spammer ID	Spammer ID
A1087DECRN5UDU	A10A0MBFUQ3V81	A10DQ97EP60BVG
A11CH0J2O9MR1Y	A11J6S7N0LIPOY	A11PXVDPJDIZY8
A12E1N994TQIH7	A12QQXGSSHWQ21	A137GQ32Q97F6C
A13CTR8SOSER5V	A13WLUIATM3NW2	A156QNPXZ736ZB
A16QM98NAYJ64	A16WM7LFPRHZ65	A16YIC86R49CQK
A16Z7G9BBYH9EN	A16ZPLT8MEVRKA	A17AY5OU3SOE3D

Table 4-26 Some of the 377 spammers detected by the second, the third, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A10ZG0QKITDBW1	A123A9IBBMYQV9	A12YJBN58HJIG
A13CADRZQEEZ23	A13CTR8SOSER5V	A13MWIWORVRQ5S
A13RVB474Y25MW	A13V1OMU4SP9H7	A13WLUIATM3NW2
A13Y57V8MRPUZN	A146QCGOQPZLTG	A14PPHRETAWA0C
A15A7XGROJQBT9	A15B5DO3A3A30C	A15IO2K28WY330

Spammer ID	Spammer ID	Spammer ID
A15LH1T332KVNR	A15LVJNNKFPR6U	A15TIQSRJFRW1W

Table 4-27 Some of the 56 spammers detected by the second, the forth, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A13CTR8SOSER5V	A13WLUIATM3NW2	A17NDDPCEDFGI6
A1838IV34H97Y7	A1AM1OTVG6LMH6	A1CGOXRZ1A9PCY
A1EU7DV8VN9MYN	A1F2MIPQVYMH9D	A1G5ODL9M1U1UH
A1KM8XML9UL517	A1P9FKUQH96W7M	A1QKX0VV7QWSRZ
A1R13RFLWY0PTY	A1WFYHLFLB0KCI	A20P0UJT8PCHU4
A244KA5N14SPXT	A247FLYIIJTC0T	A2KCQ4FSOIXU4Y

Table 4-28 Some of the 44 spammers detected by the third, the forth, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A117ZEQ3R1OE0J	A13CTR8SOSER5V	A13WLUIATM3NW2
A14QZ17AH7C4XW	A15JP7A4J0DI3P	A160WJ3ZRC0G0O
A16EG3F92ZZWQ5	A17NDDPCEDFGI6	A1838IV34H97Y7
A19HHWZTNY8X19	A19ZFHD32VMZMM	A1ALQZJ0MZ6YV
A1AM1OTVG6LMH6	A1D6TLPS9OUQ53	A1EU7DV8VN9MYN
A1F2MIPQVYMH9D	A1G5ODL9M1U1UH	A1P9FKUQH96W7M

The spammers who have been detected by four different approaches can be found in Table 4-29 and Table 4-30. These reviewers possess the highest possibility to be the spammers as they have been discovered by the largest possible number of the approaches.

Table 4-29 The spammers detected by the first, the second, the third, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A1B9ISLYYCHIVE	A1DBQURAZBR8T	A1G5Q9HBN0EGDV
A2OLJM7IREKPWZ		

Table 4-30 The spammers detected by the second, the third, the forth, and the fifth approaches

Spammer ID	Spammer ID	Spammer ID
A13CTR8SOSER5V	A13WLUIATM3NW2	A17NDDPCEDFGI6
A1838IV34H97Y7	A1AM1OTVG6LMH6	A1EU7DV8VN9MYN
A1F2MIPQVYMH9D	A1G5ODL9M1U1UH	A1P9FKUQH96W7M
A1QKX0VV7QWSRZ	A244KA5N14SPXT	A247FLYIIJTC0T
A2KCQ4FSOIXU4Y	A2KU73UCKBTGN4	A2PKDTI0FJLLY3
A2R8TI6YN6NY9C	A2S94UYOSEJGB	A2UDKE5MGN44ID
A2Y2G8EHR55H2J	A2ZDLKHY2IIRVG	A30FTTSISBP4A0
A3AMA020X8FPJS	A3N73VIVT5YVW5	A3RXLRWD0OI7O6
A3SDJDZK2FPB7U	A3T0PAEBBZFYFC	A6ARHT06FJE8O
A6NARW2TX35UO	AC2FHEJ1YDHCB	ACAT1IO5USZCV
AD43RR7GUCXKP	ADWZ4I7M2RQUD	AOYKZHA8QR31R
ARIH2IC78DIIJ	AX1PD3IN7JHOZ	A117ZEQ3R1OE0J
A14QZ17AH7C4XW	A15JP7A4J0DI3P	A160WJ3ZRC0G0O
A16EG3F92ZZWQ5	A19HHWZTNY8X19	A19ZFHD32VMZMM
A1ALQZJ0MZ6YV	A1D6TLPS9OUQ53	

Since the intersection of the results from all the proposed approaches is an empty set, the above detected spammers are the most possible ones.

The reason why there is no common reviewer among all the approaches is that the essences of the approaches are different from each other, especially the one which detects

spammers with dense regions in their timelines. Therefore, they discover independent sets of results which might not necessarily have intersections.

The reason of the occurrence of this issue in our dataset is that there was no reviewer who has made a sudden change in the trends of positive/negative ratios of a product, and in the same time has many extreme-rated reviews as well as dense regions in his timeline.

Also, there was no reviewer who has made a sudden change in the trends of positive/negative ratios of a product, and in the same time has dense regions in his timeline as well as reviews which are extreme-rated if they are his target, or rated with a close-to-mean value otherwise.

Chapter Five: Conclusions and Future Research Directions

This chapter first summarizes all the methods proposed to detect review spam, and gives a conclusion. Afterwards, some other tasks and approaches which can be done in the future to extend this work are explained.

5.1 Conclusions

Nowadays, the online reviewing systems have become so popular since most of the individuals refer to the online available reviews about the products or services they intend to purchase before actually paying for them. On the other hand, some of those who already have some experiences of the products and services are interested in publishing their opinions about them in reviewing websites to let future customers know their feelings toward them. This huge popularity of the online available reviews might result in motivating some people or companies to insert their fake reviews in the reviewing systems to mislead the consumers. Their intention might be promoting their own products or services, or defaming those of their competitors.

Although some other methods have been introduced previously to detect review spam, five more approaches have been proposed in this thesis to face the issue from three different points of view; investigating the trends of reviews, analyzing reviewers' behaviors, and detecting spammer groups.

In the first category, a method has been introduced which considers the trends of the ratios of positive, negative, and neutral reviews for each product. The method detects the intervals with abrupt changes in the computed ratios as suspicious intervals, and counts the reviews in those intervals as candidate review spam. Then, the reviewers with many candidate review spam are regarded as the spammers.

The first introduced method in the second category tries to find the reviewers a significant number of whose reviews are rated with the extreme values, i.e., the highest or lowest possible ratings, and further announces those reviewers as the detected spammers.

Based on a concept in the shilling attack domain in recommendation systems, the next approach detects those reviewers whose reviews are either extreme-rated or have rating within a distance of 1.1 from the average ratings of their corresponding products. The reviewers not even one of whose reviews are out of the above two cases will be declared as the spammers.

The last approach in this category detects spammers by considering the fact that the spammers are asked to post their fake reviews in a short time span. Therefore, the reviewers who have published a majority of their reviews in short time are highly likely to be spammers.

The proposed method in the last category intends to detect spammer groups whose members share the task of publishing review spam among each other. For each of their targeted product, one of them posts an extreme-rated review, while others publish their reviews with the same polarity but with a value close to the average rating of the product. The suggested method discovers this type of spammer groups.

In summary, there exist various attitudes in writing spam reviews which result in the necessity of the existence of various approaches towards review spam detection. In other words, there is no comprehensive solution to tackle this problem. This thesis has provided some approaches which we claim to be newly-introduced, domain-independent, and affective for discovering review spam. Furthermore, it has compared the suggested approaches, and ranked the results.

5.2 Future Directions

One worthwhile task to accomplish in this area is to build a gold-standard dataset. Since the exact spammers and spam reviews are specified in such dataset, the accuracy of the results of any suggested approach and moreover its usefulness in detecting review spam can be examined. Furthermore, all the needed data for different approaches can be provided by creating the suggested dataset.

Utilizing NLP (natural language processing) is another trend which can be applied in the future. Although it has been done extensively in the previous works in the area, I have assumed I am dealing with review spam published by expert spammers who write well-crafted reviews such that their reviews look just like ordinary reviews, and therefore they usually cannot get detected by examining their content or properties, such as review length. The future work would concentrate on integrating the methods proposed in this thesis with NLP, not to apply NLP as a standalone approach.

One other idea is that if a reviewer writes a review with positive rating on a product, and one or more reviews with negative ratings on some other product from the same category,

e.g. a positive review on a tablet produced by one company and some negative reviews on tablets produced by other companies, he has committed a suspicious act. Furthermore, if he performs such a job several times for different categories, he is a spammer candidate.

Another approach is to investigate whether a reviewer has several negative reviews on different products from the same brand, or several positive reviews on different products from another brand, or both since that person can be a spammer candidate as well.

However, both of the above cases cannot be tested using our dataset since there is no data available denoting the product types as well as the product brands.

For the datasets which contain the usefulness ratings in addition, one approach to detect review spam is to consider the usefulness feature. To do so, the people who mark reviews as useful should be investigated. For each review, if a great proportion of the individuals who mark it as useful have been detected as spammers previously, that review is highly probable to be spam as well. If a reviewer has many such spam reviews, he will be recognized as a spammer.

Another idea is to build the social network of the reviewers. The reviewers would be the nodes while the links between them will be weighted according to the closeness of their reviews ratings and the time when they have posted their reviews. Hence, for every two reviewers, for each of their common products, if the ratings of their reviews on the common product are the same and also their reviews have been published on the same day, the weight of the link between those two reviewers will be added up by the highest possible value. The concept of the normal distribution can be utilized to define the way of

calculating the weight of the link between each pair of the reviewers. Afterwards, some community detection algorithms can be employed to detect groups of spammers.

Bibliography

[1] Nitin Jindal, Bing Liu, Review spam detection, Proceedings of the 16th international conference on World Wide Web, May 08-12, 2007, Banff, Alberta, Canada.

[2] Nitin Jindal, Bing Liu, Analyzing and Detecting Review Spam, ICDM2007.

[3] Nitin Jindal, Bing Liu, Opinion spam and analysis, Proceedings of the international conference on Web search and web data mining, February 11-12, 2008, Palo Alto, California, USA.

[4] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, Hady Wirawan Lauw, Detecting product review spammers using rating behaviors, Proceedings of the 19th ACM international conference on Information and knowledge management, October 26-30, 2010, Toronto, ON, Canada.

[5] Arjun Mukherjee, Bing Liu, Junhui Wang, Natalie Glance, Nitin Jindal, Detecting Group Review Spam, Proceedings of International Conference on World Wide Web (WWW-2011, poster paper), 2011.

[6] Nitin Jindal, Bing Liu, Ee-Peng Lim, Finding unusual review patterns using unexpected rules, Proceedings of the 19th ACM international conference on Information and knowledge management, October 26-30, 2010, Toronto, ON, Canada.

[7] Fangtao Li, Minlie Huang, Yi Yang, Xiaoyan Zhu, Learning to identify review Spam, IJCAI 2011.

[8] Guangxia Li, Steven C. H. Hoi, Kuiyu Chang, Two-view Transductive Support Vector Machines, SDM, pages 235–244, 2010.

[9] Myle Ott, Yejin Choi, Claire Cardie, Jeffrey T. Hancock, Finding deceptive opinion spam by any stretch of the imagination, Proceedings of the 49th Annual Meeting of the

Association for Computational Linguistics: Human Language Technologies, June 19-24, 2011, Portland, Oregon.

[10] Sihong Xie, Guan Wang, Shuyang Lin, Philip S. Yu, Review Spam Detection via Time Series Pattern Discovery, WWW 2012 Companion, Lyon, France, April 16–20, 2012.

[11] Arjun Mukherjee, Bing Liu, Natalie Glance, Spotting Fake Reviewer Groups in Consumer Reviews, WWW 2012, Lyon, France, April 16–20, 2012.

[12] William B. Cavnar, John M. Trenkle, N-gram-based text categorization, Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, NV, 1994.

[13] Guan Wang, Sihong Xie, Bing Liu, Philip S. Yu, Review graph based online store review spammer detection, IEEE International Conference on Data Mining, 2011.

[14] Rakesh Agrawal, Tomasz Imieliński, Arun Swami, Mining association rules between sets of items in large databases, Proceedings of the 1993 ACM SIGMOD international conference on Management of data, p.207-216, May 25-28, 1993, Washington, D.C., United States.

[15] Shyong K. Lam, John Riedl, Shilling recommender systems for fun and profit, Proceedings of the 13th international conference on World Wide Web, May 17-20, 2004, New York, NY, USA.

[16] Bing Liu, Sentiment analysis and subjectivity, Handbook of Natural Language Processing, second edition, 2010.

[17] Eric Breck, Yejin Choi, Claire Cardie, Identifying expressions of opinion in context, Proceedings of the International Joint Conference on Artificial Intelligence, 2007.

[18] Paula Chesley, Bruce Vincent, Li Xu, Rohini Srihari, Using verbs and adjectives to automatically classify blog sentiment, AAAI Symposium on Computational Approaches to Analysing Weblogs, pp. 27–29, 2006.

[19] Kushal Dave, Steve Lawrence, David M. Pennock, Mining the peanut gallery: Opinion extraction and semantic classification of product reviews, Proceedings of WWW, pp. 519–528, 2003.

[20] Michael Gamon, Anthony Aue, Simon Corston-Oliver, Eric Ringger, Pulse: Mining customer opinions from free text, Proceedings of the International Symposium on Intelligent Data Analysis, pp. 121–132, 2005.

[21] Bo Pang, Lillian Lee, Shivakumar Vaithyanathan, Thumbs up? Sentiment classification using machine learning techniques, Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 79–86, 2002.

[22] Ellen Riloff, Janyce Wiebe, Learning extraction patterns for subjective expressions, Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2003.

[23] Janyce Wiebe, Rebecca Bruce, Thomas O'Hara, Development and use of a gold standard data set for subjectivity classifications, Proceedings of the Association for Computational Linguistics, pp. 246–253, 1999.

[24] Peter Turney, Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews, Proceedings of the Association for Computational Linguistics, pp. 417–424, 2002.

[25] Theresa Wilson, Janyce Wiebe, Paul Hoffmann, Recognizing contextual polarity in phrase-level sentiment analysis, Proceedings of the Human Language Technology

Conference and the Conference on Empirical Methods in Natural Language Processing, pp. 347–354, 2005.

[26] Hong Yu, Vasileios Hatzivassiloglou, Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences, Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2003.

[27] Murthy Ganapathibhotla, Bing Liu, Mining Opinions in Comparative Sentences, Proceedings of the International Conference on Computational Linguistics, 2008.

[28] Nitin Jindal, Bing Liu, Identifying comparative sentences in text documents, Proceedings of the ACM Special Interest Group on Information Retrieval, 2006.

[29] Nitin Jindal, Bing Liu, Mining comparative sentences and relations, Proceedings of American Association for AI National Conference, 2006.

[30] Zoltan Gyongyi, Hector Garcia-Molina, Web Spam Taxonomy, Tech. Report, Stanford University, 2004.

[31] Baoning Wu, Vinay Goel, Brian Davison, Topical TrustRank: using topicality to combat Web spam, WWW, 2006.

[32] Barry Leiba, Nathaniel Borenstein, A multifaceted approach to spam reduction, In Proceedings of the First Conference on Email and Anti-Spam, 2004.

[33] Isidore Rigoutsos, Tien Huynh, Chung-kwei: a pattern-discovery-based system for the automatic identification of unsolicited e-mail messages (spam), In Proceedings of the First Conference on Email and Anti-Spam, 2004.

[34] Paul Resnick , Hal R. Varian, Recommender systems, Communications of the ACM, v.40 n.3, p.56-58, March 1997.

[35] Myles Anderson, http://searchengineland.com/study-72-of-consumers-trust-online-reviews-as-much-as-personal-recommendations-114152