Expert Systems With Applications 58 (2016) 83–92

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Detection of fake opinions using time series

Atefeh Heydari*, Mohammadali Tavakoli, Naomie Salim

Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia

ARTICLE INFO

Article history:
Received 2 September 2015
Revised 29 January 2016
Accepted 6 March 2016
Available online 24 March 2016

Keywords:
Review spam
Spam detection
Opinion spam
Fake reviews

ABSTRACT

Today's e-commerce is highly depended on increasingly growing online customers' reviews posted in opinion sharing websites. This fact, unfortunately, has tempted spammers to target opinion sharing websites in order to promote and demote products. To date, different types of opinion spam detection methods have been proposed in order to provide reliable resources for customers, manufacturers and researchers. However, supervised approaches suffer from imbalance data due to scarcity of spam reviews in datasets, rating deviation based filtering systems are easily cheated by smart spammers, and content based methods are very expensive and majority of them have not been tested on real data hitherto.

The aim of this paper is to propose a robust review spam detection system wherein the rating deviation, content based factors and activeness of reviewers are employed efficiently. To overcome the aforementioned drawbacks, all these factors are synthetically investigated in suspicious time intervals captured from time series of reviews by a pattern recognition technique. The proposed method could be a great asset in online spam filtering systems and could be used in data mining and knowledge discovery tasks as a standalone system to purify product review datasets. These systems can reap benefit from our method in terms of time efficiency and high accuracy. Empirical analyses on real dataset show that the proposed approach is able to successfully detect spam reviews. Comparison with two of the current common methods, indicates that our method is able to achieve higher detection accuracy (F-Score: 0.86) while removing the need for having specific fields of Meta data and reducing heavy computation required for investigation purposes.

1. Introduction

With the development of internet, people became more confident to explain their thoughts on websites and share them with millions of people (Heydari, Tavakoli, Ismail, & Salim, 2016). Web 2.0 slowly changed different aspects of people living. For instance, by creating online groceries, a huge number of daily trades are virtualized.

Nowadays people are more dependent to the internet for purchasing products and services. Long time ago, when they wanted to purchase a product, the best method was asking other customers who have purchased it before and know about the quality of that product very well to ensure that they will have a successful transaction.

Similarly, now they can visit customer reviews about various products or services that they tend to purchase via opinion sharing websites. Hence they can easily trade off the pros and cons of a specific good. The increasingly propensity of people to use online opinion sharing websites has created a challenging situation for manufacturers, business holders and stores (Peñalver-Martinez et al., 2014). Hence dishonest producers who tend to control and optimize the customers' opinions flow on their products and brand attempt to publish fake reviews among review websites. Sometimes they hire individual or in some cases groups of spammers to create not only glamorized positive reviews on their products but also harmful negative reviews on competitors'. These types of non-truthful reviews motivate customers to find their products the best option to purchase among similar products offered by other brands.

Fake opinions are extremely harmful not only for potential customers but also for business holders. Therefore, opinion mining techniques are assisting business to analyze posted customers' opinions on offered products to detect and filter spam reviews and proffer truthful reviews to purchasers (Savage, Zhang, Yu, Chou, & Wang, 2015). However research in this area is not adequate and many critical problems related to spam detection are not solved yet.

A bunch of previous approaches relied on content based factors to detect spam reviews (see Section 2). In an approach proposed by Lim, Nguyen, Jindal, Liu, and Lauw, (2010), for example, the

* Corresponding author. Tel.: +989212378119/+60197623624.
E-mail addresses: hatefeh2@live.utm.my, a_tav_ir@yahoo.com (A. Heydari), tmohammadali2@live.utm.my (M. Tavakoli), naomie@utm.my (N. Salim).
http://dx.doi.org/10.1016/j.eswa.2016.03.020
0957-4174/

product features mentioned in a review are compared with other reviews to identify duplicate reviews and filter them as spam. Although they are applicable on any type of reviews, content based methods naturally need expensive computations.

Other approaches focused on rating behaviors (Algur, Patil, Hiremath, & Shivashan, 2010) or/and other available Meta data of reviews (see Section 2). Majority of these methods require certain features that are included in a few number of datasets. However, most of these features, such as rating, author's ID, and helpfulness, could be manipulated perfectly by spammers to appear as real opinion.

Our approach differs significantly from former studies in several manners. Firstly, our method narrows down the selection of candidates for textual similarity investigation by constructing time series of reviews for each product and capturing only suspicious time intervals. This novation removes the need of expensive comparisons. Secondly, spam reviews generated by spammers who try to mislead customers without exhibiting any anomalous rating behavior are easily detectable by our method. This is because our approach does not merely focus on rating behaviors for detection of spam reviews. Finally, there are a few number of widely available fields of Meta data required in our method making it comprehensively applicable on different review websites and datasets.

The contribution of this paper is (1) To testing the suitability of using time series analysis approaches accompanied with a synthetic spam scoring system for detection of spam reviews and, consequently, developing a robust review spam detection system, (2) To reduce the need for expensive computations of detection phase by narrowing down the selection of samples.

The remainder of the paper is structured as follows: the next section discusses related work. Section 3 discusses the proposed method for detection of spam reviews. Experimental analysis are presented in Section 4. Finally, Section 5 presents our conclusions and future work.

2. Related work

In comparison with other types of spam such as e-mail spam (Wu, Feng, Wang, & Liang, 2015), web spam (Fdez-Glez et al., 2015), and SMS spam (Ahmed et al. 2015), detection of spam reviews is very nontrivial because manual evaluation of reviews and distinguishing fake reviews from real opinions is almost impossible (Jindal et al. 2008). Hence, state-of-the art methods in detecting various types of spam are not applicable in this domain. Accordingly, detection of spam reviews could be considered as one of the sophisticated problems in Natural Language Processing domain. A comprehensive review of state-of-the-art approaches in detection of spam reviews is provided in our previous research (Heydari, ali Tavakoli, Salim, & Heydari, 2015). These approaches can be broken down into the three categories of detecting group of spammers, detecting spammers, and detecting spam reviews:

(A) Detection techniques for group spammers

Some of the spam attacks are organized by group of spammers and a part of spam detection approaches have focused on detecting group of spammers, though the number of these approaches is limited. Mukherjee, Liu, and Glance, (2012),and Mukherjee, Liu, Wang, Glance, and Jindal (2011) defined diverse group spam indicators to detect group spammers such as rating deviation of members of a group of spammers, content similarity between group members, and number of products for which the group is creating spam reviews. By the construction of a relational model, the authors used the relationship between groups, individuals and products to score candidate groups. Similar relational models and features were used latter in Kolhe, Joshi, Jadhav, and Abhang (2014). Although both of the studies considered textual similarity of reviews as a spam sign, Ye and Akoglu, (2015) only used a graph-based measure to find statistical distortions caused by spamming activities and cluster the groups of spammers.

(B) Spammers detection techniques

Graph-based approaches consisting of graphs with review, reviewer, and store nodes (Akoglu, Chandy, & Faloutsos, 2013; Fayazbakhsh & Sinha, 2012; Wang, Xie, Liu, & Yu, 2011) focused mainly on using rating behaviors of reviewers to detect spammers. Rating deviation was one of the main features (Lim et al., 2010; Mukherjee et al., 2013; Sharma et al. 2013) or the only feature (Akoglu et al., 2013; Aye & Oo, 2014; Jindal, Liu, & Lim, 2010; Savage et al., 2015; Xue et al., 2015) used in detection of opinion spammers.

One of the indicators of the quality and fame of a product is its rank obtained from reviews. Thus, distortion of product's rank is one of spammers' main targets. However, with observation of online reviews, it could be seen that in some spam reviews, given rate is incompatible with the content. It shows that spammers are conscious about filtering technologies and try to pass through rating deviation-based filtering systems. They rate a product moderately, while trying to mislead customers by their words. We alleviate this problem in our approach by taking into account the context of reviews and activeness of a reviewer in every captured suspicious time interval. With this method, spam reviews of not only rating deviators, but also smarter spammers would be captured.

With the goal of detecting singleton spammers, the study carried out by Xie, Wang, Lin, and Yu (2012) was focused on reviewers' behaviors. A singleton reviewer is a reviewer who has written only one review. The authors assumed that reviewers' behaviors can be divided into two phases: arrival phase: when a customer purchase a product or a spammer get hired, and writing phase: when they start developing reviews. The authors analyzed spammers and customers' behaviors in normal arrival, promotion arrival and spam attack arrival. Accordingly, they found that spammers start writing phase immediately after arrival but customers have delay for receiving product and testing it. Therefore, the authors focused on nonstandard patterns in arrival phase to do their task. Consequently, in another method proposed in Fei et al. (2013) posting time of reviewers were used to detect spammers. The authors generated 5 new spammer behavioral features as indicators to be used in review spammer detection. Their method reveals more accurate results comparing to Xie et al., (2012). However, one of these 5 features is 'Ratio of Amazon verified purchase', a rarely available feature, which possibility of using this feature in any detection technique optimizes the accuracy of the method profoundly.

In order to develop a comprehensive detection system, utilized features should be general to enable the proposed system to work in disparate circumstances and on different datasets. The Amazon verified purchase sign indicates that the reviewer has really purchased the target product and the probability of being a spammer for him/her is almost zero. Although the work of Fei et al., (2013) was successful in detecting spammers, it could be used in a limited number of datasets. In contrast, our system uses effective features acquired from processing abundantly available data that could be found in almost all of the product review datasets. Moreover, our scope is to detect all types of spam reviews, including singleton, multiplton, advertisements, random texts, rank promoters, and fake reviews.

(C) Spam review detection techniques

The highest proportion of state-of-the-art techniques and methods in the field are proposed to detect and filter spam reviews (Tavakoli, Heydari, Ismail, & Salim, 2015). Plenty of these studies tried to demonstrate how spam reviews differs from real opinions in terms of sentiment and linguistic aspects (Ott et al., 2011, 2012; Deng, and Chen, 2014; Feng, Banerjee, & Choi, 2012; Long et al., 2014; Peng & Zhong, 2014;), writing style (Banerjee & Chua, 2014; Fusilier, Montes-y-Gómez, Rosso, & Cabrera, 2015), and subjectivity and readability (Ong, Mannino, & Gregg, 2014). Majority of these approaches have been conducted on synthetic datasets initially introduced by (Ott et al., 2011). However, by performing same methods on synthetic and real datasets, (Mukherjee et al., 2013; Morales et al., 2013) argued that synthetic datasets are defective. Thus, development and evaluation of detection techniques based on these synthetic datasets can be problematic, as they do not appropriately reflect real world review spam (Crawford, Khoshgoftaar, Prusa, Richter, & Najada, 2015). Another drawback of these studies is that the spammers can modify their language to mimic genuine users as closely as possible and avoid detection. Finally, there are many spammers writing their genuine experience about a really purchased product for a non-purchased product in order to spam it (e.g. the spammer has a Canon camera and write positive spam reviews for Nikon camera based on his experience of Canon camera). In these common cases, taking context of reviews into account is not efficient any more.

Although lack of reliable evaluation calls the accuracy of content based studies into question, a bunch of approaches demonstrated that focusing on context similarity of reviews is advantageous. In these approaches, duplicate and near duplicate reviews were considered as spam (Algur et al., 2010; Jindal et al. 2007; Jindal et al. 2008; Lau et al., 2011; Lin et al., 2014; Shashirekha, Murali, & NAgabhushan, 2009). Content similarity comparison is a famous technique among researchers as it is generally believed that spammers create a few number of fake reviews and try to copy it in different situations, with various identities and for diverse products of a brand. However, all these methods exhibit a key problem: Investigation of similarity among all reviews requires time-consuming comparisons and a huge number of assessments is required in most cases. To overcome these drawbacks, in our experiment, suspicious time intervals are captured and similarity of contents is only assessed among reviews fallen in these intervals. It declines the number of comparisons and, consequently, increases the speed of the system dramatically. Moreover, there are many short reviews that are frequently used by reviewers such as “Good product, good price”. Current similarity based systems capture majority of these reviews as duplicate and spam. We alleviate this problem by assessing similar reviews in a short period of time, suspicious interval, whereas duplication of such reviews is unlikely to be haphazard. In addition, our approach is a combinatorial method whereby even duplicate reviews have to obtain adequate fakeness scores from two other assessments to be identified as fake reviews.

3. Spam detection framework

In this section, we first illustrate some definitions and abbreviations that are used throughout the paper (see Table 1). Then we describe the model which constructs time series for each product and uses detection metrics to distinguish spam and non-spam reviews (see Section 3.2). The brief flowchart of our algorithm is shown in Fig. 1. And we explain how the system uses detection metrics to detect spam reviews (see Section 3.3).

Table 1
Symbols used in constructing time series.

Symbol	Explanation
b	brand
b_i	Review i written for brand b
d_i	Date of posting review i
n	Number of reviews
Δt	Size of time window
Len	duration of time between lunching products of a brand and last review
Int_n	nth time interval in the time series
N	Last interval in a time series
R(Δt)	Set of reviews placed in each Δt
R(b)	Rank of brand b
$AVG(Int_n)$	Average number of reviews in Int_n
$RN(Int_n)$	Number of reviews in Int_n
$CS(b_i, b_j)$	Cosine Similarity method to calculate similarity of reviews b_i and b_j
$S(b_i)$	Score of similarity of review b_i

3.1. Definition

In this paper, we aim to propose a system which is capable to detect spam reviews effectively. The system constructs time series of number of reviews for each brand and distinguishes spam reviews from real opinions after detecting suspicious intervals.

3.2. Model description

The traditional review spam detection techniques ignore the burst patterns of reviewing the products as a significant evidence of spam attacks. They just employ a number of features to detect spam reviews. In fact, when spammers start to generate fake reviews for a product, number of reviews will raise in that certain interval because these spam reviews will be added to the usual truthful reviews in the interval and create a burst. Construction of our review spam detection system is based on this consideration as the burst patterns identification techniques are used to identify suspicious time intervals and fakeness of reviews are investigated across them. A brief flowchart of our method is shown in Fig. 1.

We accumulated reviews of different products of a certain brand to form the time series for the brand. This is because spammers usually generate fake reviews to promote or demote a brand. Moreover, in many products, number of reviews are not enough to construct a time series. Brand b has a set of reviews $\{b_1, b_2, \ldots, b_n\}$ that each one has a corresponding date shown by d_i. Duration of time between launching a brand and last review (len) is $\{d_n - d_1\}$. Thus, axis X of the time series would be dates (from d_1 to d_n) and the Y axis will show number of reviews in d_i. A time window size (Δt) is defined here to divide the element of time into equal intervals (Int). Various sizes are considered as Δt in current research in order to obtain the most accurate result. These window sizes are defined as 7days, 14days, 21days and 30 days in this study.

$$\text{Number of } (Int\,jlv) = \frac{len}{\Delta t} \tag{5}$$

Average number of reviews in an interval $AVG(Int_i)$ could be calculated as follow.

$$AVG(Int_i) = \frac{n}{\text{Number of } (Int)} \tag{6}$$

The time window is defined to be slid on time series in order to capture burst patterns in review process. Accordingly, two template patterns ($temp$), showing a peak in a curve, were defined to

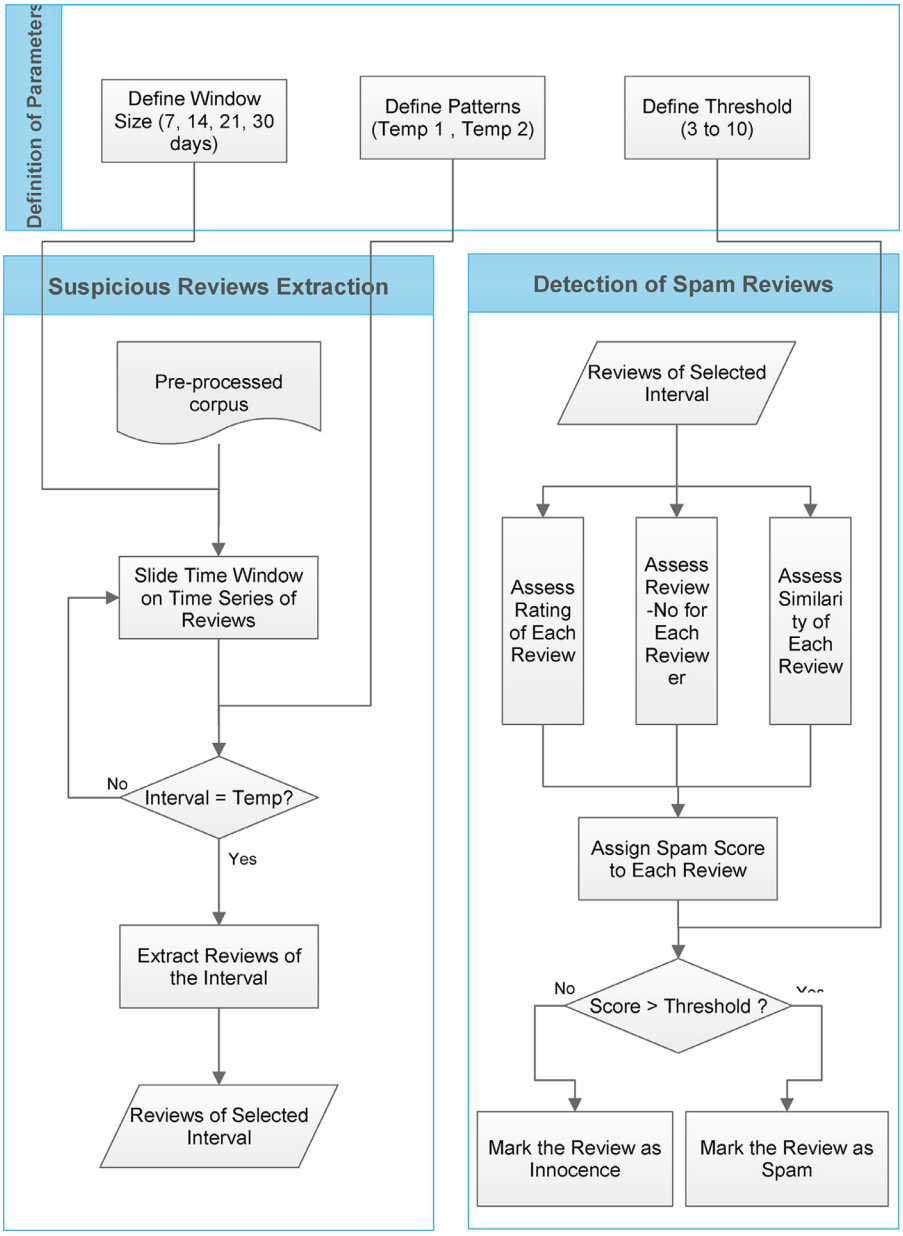

Fig. 1. The flowchart of the method.

capture intervals containing peak points. The patterns were defined as:

$$\text{temp1} = \{(I-2)_i < I_i,\ (I-1)_i\ \langle I_i \rangle\ (I+1)_i,\ I_i > (I+2)_i\} \quad (7)$$

$$\text{temp2} = \{(I-1)_i < I_i > (I+1)_i\} \quad (8)$$

Where I_i is the number of reviews in interval (I) which is compared with neighbor intervals to detect peak-points. The reason of considering two different patterns is dissimilar covering areas of each one which cause various accuracy levels for the method. The duration of time that Δt could be slid on is $[d_1, d_{n-\Delta t})$. Set of reviews placed in each Δt is

$$\mathrm{R}\ \Delta t = \{b_j | d_j \in Int_i\} \quad (9)$$

After sliding time window on the curve of each brand and performing template pattern finding (Fig. 2), captured intervals with associated reviews are stored in a list. Then, the intervals with $\mathrm{R}\,\Delta t < AVG(Int_i)$ are discarded for normal flow of reviewing in them. The final list comprises reviews of suspicious burst intervals.

Many of the captured burst patterns might be due to seasonal fluctuations and promotions. Thus, detection metrics (see Section 3.3) are then used to score each review. Ultimately, reviews with spam scores greater than the defined threshold are marked as spam.

3.3. Detection metrics

As we mentioned before, the burst patterns might be due to promotional or seasonal reasons. To confirm that the suspicious increment in number of reviews at captured intervals is for spammers' activities and to conserve real opinions fallen in suspicious intervals, we employ following criteria to score each review and detect spam ones.

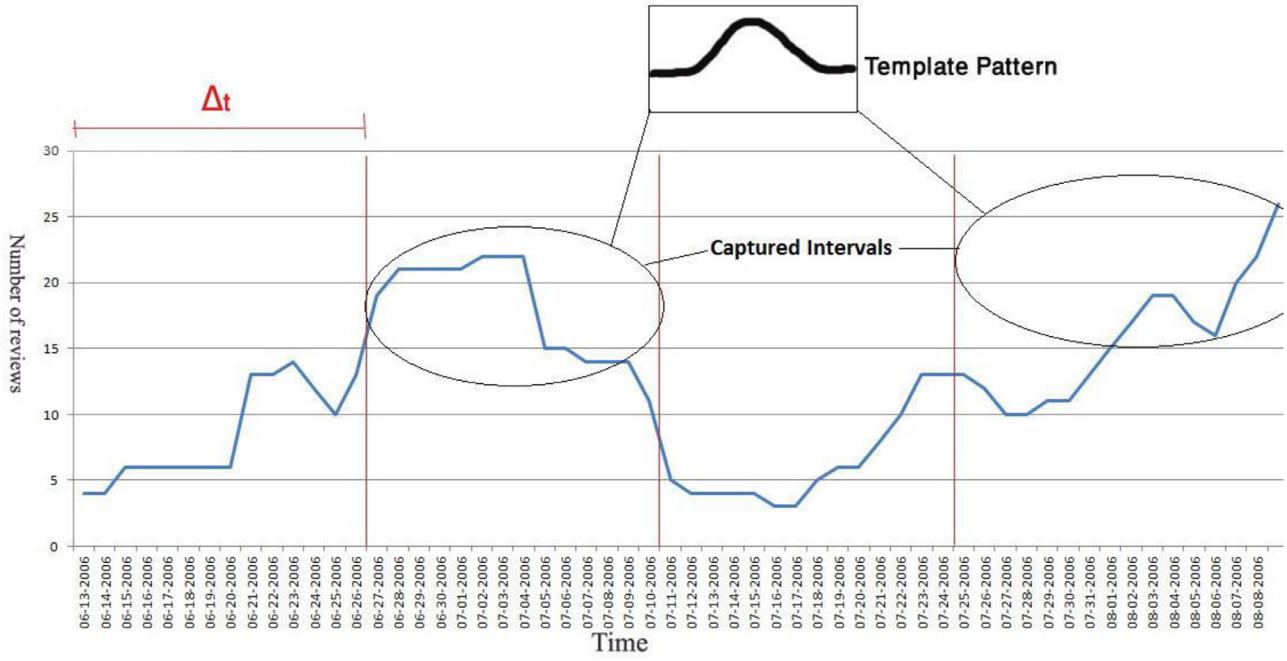

Fig. 2. Capturing peak intervals in number of reviews.

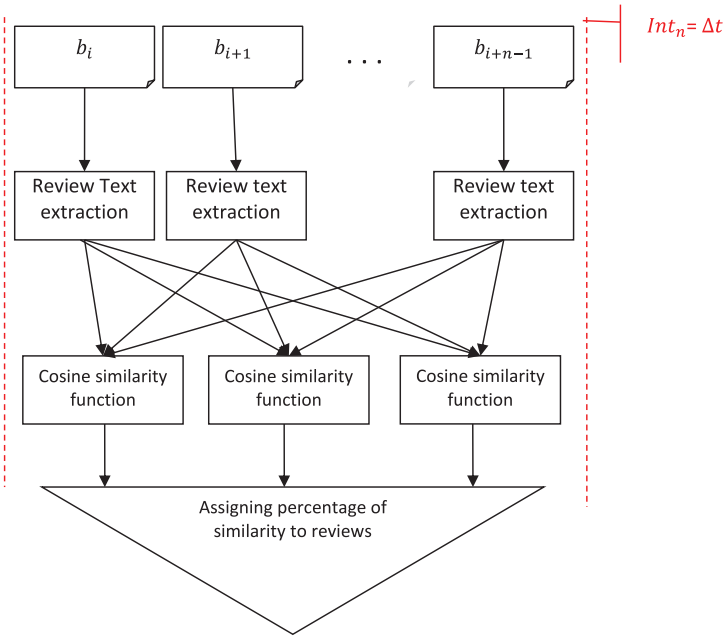

Fig. 3. Calculating content similarity between reviews of a time section.

3.3.1. Context similarity

It is generally accepted that a spammer usually uses a certain vocabulary to generate fake reviews. So, his fake reviews are very close to each other in terms of utilized vocabulary. Based on this consideration, we assigned an spam score to the reviews of each captured interval by measuring their similarities with other reviews of the interval. Given two term frequency attribute vectors A and B associated with *docA* and *docB*, and according to Euclidean dot product formula ($a.b = ||a||\ ||b||\ \cos\theta$) the similarity could be computed as:

$$\cos(\theta) = \frac{A\ .B}{||A||\ ||B||} = \frac{\sum_{i=1}^{n} A_i * B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} * \sqrt{\sum_{i=1}^{n} (B_i)^2}} \quad (13)$$

Where θ is the angle between the gradients of *docA* and *docB*. The cosine measure of similarity was used between texts of reviews fallen in a suspicious interval. Fig. 3 illustrates the way of using cosine similarity in current research to measure the similarity of reviews fallen in a suspicious interval and score them.

Correspondingly, the percentage of similarity for the review b_i could be calculated by the following formula.

$$S(b_i) = \sum_{di\epsilon\ Int_n} \sum_{j=i+1} CS(r_i, r_j) \quad (14)$$

3.3.2. Authors' activeness

In this step number of reviews of an author fallen in a certain suspicious interval will be counted. Then a spam score will be

Algorithm 1: Algorithm for calculating number of reviews for a person that are fallen in a peak interval.
Inputs : List of Reviews fallen in a specific suspicious interval

```
FOR each review( i ) of the list
{
    FOR each reviews ( j ) of list bating review ( i )
    {
        IF (Reviewer_i == Reviewer_j)
            Counter + +
    }
Review-no of review( i ) = Counter
Counter = 0
}
```

Fig. 4. The algorithm of calculating number of reviews written by a person in an interval.

Algorithm 2: Algorithm for calculating deviation of a review rate.
Inputs : List of all Reviews, List of Reviews fallen in a specific suspicious interval

```
FOR each review( i ) of the list (all reviews)
    {
        Ranks = Ranks + Rate of Review( i )
    }
Brand Rank = Rank / Number of all reviews
FOR each review ( i ) of the interval list
    {
        Deviation of Review( i ) = Rate of Review( i ) - Brand Rank
    }
```

Fig. 5. The algorithm of calculating rating deviation of reviews.

assigned to them based on their proliferation. In the other words, an author with more number of reviews in an interval, receives greater spam score for his reviews. Hence, given a reviewer *Re*, reviews written by him could be $\{re_1, re_2, \ldots, re_n\}$. Accordingly the number of his reviews fallen in the peak-point captured interval $Re(Int_i)$ will be:

$$Re(Int_i) = \{re_j | d_j \in [(i-1) \times \Delta t, i \times \Delta t), \quad i \in \{1, \ldots, \text{Number of } (Int)\}\} \tag{10}$$

$Re(Int_i)$ is computed for reviews of all captured intervals and a spam score is assigned to each review accordingly. The algorithm of calculating $Re(Int_i)$ is illustrated in Fig. 4.

3.3.3. Authors' rating behavior

Obviously, spammers' objective is to manipulate the overall rank of a product or brand. To achieve this, they have to produce fake reviews with rates that are, profoundly, deviated from the product or brand overall rank. Accordingly, the rate given to the brand in a review fallen in a suspicious interval will be a critical point for detecting spam reviews. Given a set of ratings $\{rb_1, rb_2, \ldots, rb_n\}$ allocated to a brand by reviews, the overall rank of the brand could be:

$$\text{Rank}(b) = \frac{\sum_{i=1}^{n} rb_i}{n} \tag{11}$$

Therefore, the deviation of a review fallen in a suspicious interval will be:

$$\text{Dev}(b_i) = |\text{Rank}(b) - rb_i| \tag{12}$$

The algorithm of calculating deviation of rates of reviews from product rank is represented in Fig. 5. Another spam score is assigned to each review in this phase. Reviews with more intense deviation from brand's rank will get higher scores.

4. Experimental analysis

In this section, we conduct experiments on real dataset to validate the effectiveness of our approach. The proposed approach is implemented in eclipse. All the experiments are conducted on a Linux virtual machine with Intel processors (2.00 GHz) and 4 GB memory.

4.1. Dataset

The dataset used in this research is a customized version of the collection of customer reviews crawled from Amazon.com. We downloaded the Amazon product review dataset from http://cs.jhu.edu/~mdredze/datasets/sentiment/index2.html (see Blitzer, Dredze, & Pereira, 2007). The corpus was in the form of Xml files

```
<review>
<unique_id>1432</unique_id>
<product_name>Nikon Coolpix P3 8.1MP Digital Camera with 3.5x Vibration
Reduction Optical Zoom (Wi-Fi Capable): Camera and  Photo</product_name>
<helpful>2 of 2</helpful>
<rating>5.0</rating>
<title>Coolpix P3 Camera</title>
<date>January 3, 2007</date>
<reviewer>David G. Hooper "dhooper3"</reviewer>
<reviewer_location>Pennsylvania</reviewer_location>
<review_text>Great camera with many useful features.  No USB cable was
included however which has led to a 4 week chase and no results yet
from Amazon, Nikon, or Cameta Camera, the supplier.  Poor followup in
this case.  Now I understand that Nikon no longer supplies this cable.
</review_text>
<annotation>0</annotation>
</review>
```

Fig. 6. A sample of reviews in the dataset.

Algorithm 3 : Algorithm for extracting reviews of Nikon brand among other reviews in the category of 'Cameras and Photos'

FOR each review R_i in the raw review database
{
  IF (Product name of R_i doesn't contain "Nikon")
    remove R_i from the dataset
}
Save final dataset in Xml file

Fig. 7. Nikon reviews extraction, the algorithm.

Table 2
Meta data of reviews provided in the dataset.

Unique ID	The ID number of the review
Product Name	Name of the target product
Helpfulness	Number of (un)helpful votes for the review
Rating	Number of stars given to target product
Title	Title of the review
Date	Date of posting the review
Reviewer	Reviewer's name
Reviewer's location	Reviewer's location
Review text	Content of the review

(please see Fig. 6). Obviously, such Xml files might contain characters, such as '&', '0 × 01a' and '0 × 01c', and null tags that cause errors in the program or text engineering applications. For this reason the characters were searched and replaced with acceptable ones. And a unique String was assigned to each null value.

Cameras and photos category was selected among the wide range of categories in the dataset, including books, computer and videos, cell phone and services, and software. Our method is capable to be used with any set of product or brand reviews but, for illustration purposes, reviews of Nikon brand are selected as the main corpus of this study. By the use of StAX parser, we read all of the reviews and selected those posted for Nikon products by searching the 'Nikon' word in Product Name tags and the rest discarded because their existence was undue. (Fig. 7).

Reviews in the dataset consist of a number of components listed by Table 2. However, by considering the circumstances of our approach and in order to have a comprehensive method able to work with different situations, we used the review text, rating, reviewer's name, and date fields.

As it is explained in former sections, one of the tasks to be performed on the dataset is similarity detection. To this end, in preprocessing steps, we used Porter Stemmer to stem the reviews as well as tokenizing them. Stop words were also removed using the default stop word list of the English Analyzer: A tool provided in Apache Lucene library. After removing all reviews with missing fields, this dataset consists of 673 reviews of 17 products. A summary statistic of the customized dataset is outlined in Table 3.

Table 3
Summary statistics of utilized corpus.

Name of the brand	Nikon
Number of products	17
Number of reviews	673
Overall rank of the brand	4.2358932
First review date	01-Jan-2006
Last review date	27-06-2007
Number of annotated spam reviews	53
Number of annotated real opinions	620

4.2. Evaluation metrics

In the experiment, we use the popular metrics, precision and recall, to measure the detection quality of the proposed approach. For evaluation purposes, the corpus of this study was annotated by two experts in the area of review spam detection. The evaluators had made partially different judgments. To achieve a gold standard annotated corpus, we encountered the reviews with two votes as spam and the rest as real opinions (Table 2). The precision and recall are defined as follows:

$$Recall = \frac{|R(s)| \cap |T(s)|}{|R(s)|} \tag{14}$$

$$Precision = \frac{|R(s)| \cap |T(s)|}{|T(s)|} \tag{15}$$

Where $R(s)$ denotes the number of annotated spam reviews, and $T(s)$ denotes the number of spam reviews detected by our method. As a final point, F-Measure is used to combine the Precision and Recall:

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{16}$$

4.3. Comparison

In this section, in order to show the performance improvement of the proposed approach, we compare our approach with two common methods: rating behaviors (Lim et al., 2010), and Matrix of product features (Algur et al., 2010). Although several related works are available, we could not access their annotated datasets. And implementing them is impossible with our datasets specifications and available meta-data. To implement the method proposed by Lim et al., (2010), we defined several scores (i.e. Targeting product, Targeting group, General ranking deviation, Early rating deviation) and assigned them to the reviews based on instructions explained by the authors. Their approach was to detect spammers. However, we customize it by assigning spam scores to each review to detect spam reviews. In the other hand, implementation of the second approach, constructing matrix of product features and marking duplicate reviews as spam, was more straightforward.

4.4. Experimental results

The experiment is conducted on reviews of the Nikon brand and three different approaches. Here, we conduct the experiment under the same situation to compare the accuracy of the approaches. Table 4 illustrates the accuracy of the mentioned approaches. In this table, we have reported the results revealed by using the best configuration for our approach. Influence of each parameter is discussed in Section 4.5. From the Table 3 we can observe that our proposed method outperforms the other approaches.

Table 4
The accuracy of the approaches.

	Precision	Recall	F-Score
Matrix of product features	0.54	0.6	0.56
Rating behaviors	0.73	0.78	0.75
Our approach	0.82	0.88	0.86

4.5. Impact of parameters

There are several parameters used in this experiment influencing the accuracy of the method. Firstly, utilization of different time windows to be slid on the time series shows fluctuations in the results. It approves the significance of selecting an appropriate time interval. Secondly, we define two different patterns indicating unexpected rise in the curve of reviews in order to compare the ability of detecting bursts by the use of each pattern. Finally, the foremost parameter that has a radical role in marking reviews as spam or real opinion is the threshold. By definition of an appropriate threshold, the accuracy of the method could increase significantly. The performance of our approach with various defined values for above parameters is outlined in Table 5 (figures are showing the F-score). Pertaining to detection metrics, we used a combination of all metrics in this section as it reveals the best result (see Section 4.6). Bolded figures in the table indicate greatest results.

4.6. Impact of detection metrics

In this section, we compare the influence of each detection metric (see Section 3.3) on the accuracy of the experiment. We ran the experiment with different combinations of the metrics and outlined the results in Table 6. The cells highlighted in bold are indicating the greatest values. The best configuration as explained in the former section was used to achieve the following results.

4.7. Performance analysis

Pertaining to detection metrics, as the results show, our metrics are quite significant comparing to the variety of metrics used in the literature, including length of review, frequency of brand name, product name, capital letters, and numerals in the review, and position of review, to name but a few. The critical role of these metrics in leaking fake reviews has encouraged human judges to consider these factors in their decisions. Thus, using them in detection systems would be beneficial. The other factor that boost effectiveness of these metrics is investigating them on the reviews fallen in suspicious intervals. Capturing burst patterns as indicators of spam attacks has a significant effect on accurately detecting spam reviews. Although similarity and rating deviation factors are also used in (Algur et al., 2010; Lim et al., 2010), the role of these metrics is more significant in our approach. Table 5 shows that focusing merely on one metric does not reveal high accurate results comparing to combination of them that boost the throughput of the detection system significantly.

Moreover, according to the results of the experiments, performance of the system with the time window size of 7 days is superior to other configurations. This finding unfolds the most usual period of time that the spammers spend on generating fake reviews.

Unfortunately, lack of gold standard real datasets is an obstacle for researchers in this domain to evaluate and compare their approaches precisely. This problem is for uncertainty of some types of spam reviews for annotators. However, we used a unique annotated dataset for the three approaches and, according to the results, our proposed method could improve the accuracy significantly.

5. Conclusion and future work

In this paper, a robust review spam detection system is proposed. A detailed literature survey has shown potential of the timing element when applied to this domain and lead to the development of review spam detection approach based on time series analysis methods. Based on the consideration that the capture of burst patterns in reviewing process can improve the detection accuracy, in this experiment, we propose a review spam detection approach which investigates bogusness of reviews fallen in suspicious time intervals. We employ Authors' activeness and rating behaviors as well as context similarity of reviews in each captured interval to assign spam scores to the reviews and distinguish fake reviews from real opinions. To detect burst patterns, we firstly construct a time series to assess oscillations in number of reviews for each brand. Suspicious intervals are then captured by sliding the time window on it and finding the pattern. The experiments on real dataset show that our approach outperforms the other popular traditional methods.

On the practical front of the implications offered by the findings of this paper, scientists can benefit from tradeoffs of this paper to find effective methods and features to be used in their research. Besides, the proposed system could be used by data mining experts in order to distinguishing fraudulent reviews from innocent opinions and refining their datasets of customer reviews. Review spam filters installed on opinion sharing websites also could be equipped with our approach to achieve higher accuracy in filtering deceitful reviews.

Similar to state-of-the-art approaches in the domain that have used real datasets, our approach encountered several limitations. The most important one is that it suffers from the lack of 100% accurately annotated gold-standard dataset as manually detection of spam reviews is an indefinite task. This accuracy in synthetic datasets is 100%, though using them causes a false increment in the accuracy of proposed methods and is not compatible with our approach. Another limitation of our work is that although a negligible number of spam reviews exist in non-captured intervals in few cases, the focus of our method is only on suspicious time intervals which, instead, alleviates the problem of expensive computations in scoring phase.

Apart from unavoidable limitations, revealed results indicate that the accuracy of our method is significantly high and that its proficiency is reliable to be used in real world situations. Thus, with some customizations the method could be included in modern review spam filtering systems. The need of expensive calculations in detection phase is minimized in our work which is a vantage point for real time filtering systems. Besides, private Meta data of reviews possessed by the host opinion sharing websites, such as MAC and IP address of reviewers, their click stream and log information, have not been used in our method for inaccessibility of them. These features could boost the performance of spam detection methods significantly. In practical real situations, our method could benefit from these features provided by an opinion sharing website and equip it with a solid spam filtering system.

According to the extent literature review, few studies have focused on detection of group of spammers, while these groups are highly disruptive. By considering the integral role of time component in the detection of burst patterns in spamming behaviors, a suggestion for future research might be investigation of burst patterns and identification of relations between members to detect group of spammers.

One of the significant problems in review spam detection domain is uncertainty of spam reviews causing annotators to decide in many vague situations. Thus, proposed techniques and methods suffer from the lack of gold-standard datasets. To overcome this

Table 5
The impact of the parameters (i.e. time window, template pattern, and threshold).

	Time window							
		7 days		14 days		21 days		30 days	
		Temp 1	Temp 2	Temp 1	Temp 2	Temp 1	Temp 2	Temp 1	Temp 2
Threshold	3	00.72	00.8	00.74	00.72	00.66	00.65	00.64	00.62
	4	00.74	00.82	00.77	00.74	00.7	00.68	00.66	00.63
	5	00.75	00.83	00.79	00.76	00.73	00.7	00.69	00.67
	6	00.75	00.83	00.8	00.77	00.75	00.72	00.71	00.69
	7	00.76	**00.86**	**00.84**	00.79	00.79	00.76	00.74	00.72
	8	00.64	00.71	00.69	00.67	00.63	00.6	00.59	00.58
	9	00.51	00.67	00.69	00.68	00.64	00.62	00.6	00.58
	10	00.52	00.61	00.69	00.69	00.65	00.64	00.61	00.6

Table 6
The impact of the detection metrics.

Detection metric(s)	Precision	Recall	F-score
Authors' activeness	0.6	0.67	0.63
Authors' rating behavior	0.63	0.54	0.58
Context similarity	0.67	0.7	0.68
Activeness + Rating	0.66	0.09	0.16
Activeness + Similarity	0.78	**0.88**	0.82
Rating + Similarity	0.71	0.52	0.6
Activeness + Rating + Similarity	**0.82**	**0.88**	**0.86**

problem, a suggestion might be ranking reviews from real opinion to spam based on suspiciousness.

Although the focus of our method is only on suspicious time intervals, it is able to identify fake opinions significantly as the number of fake reviews in normal reviewing process is negligible. However, in order to have a more comprehensive detection system, we believe that our method should be combined with other novel approaches. The final system would be able to assess dishonesty of reviews from different perspectives and to cover every single review across the dataset. In addition to combination of methods, another future work is investigating the impact of other key features, such as linguistic, relations of spammers, and spammers' profiles, on our approach to enhance its throughput.

To date, the problem of spamming product reviews is still open to researchers. Every proposed detection approach suffers from certain drawbacks preventing it to identify all of the harmful spam reviews. Moreover, many of these approaches require a ready set of reviews to detect spam ones. And the literature shows that the others are not as reliable as to be confidently used in real situations. Consequently, harmful spam reviews can disturb many potential customers before being detected and filtered. Therefore, transformation from detection to prediction of spamming activities would be our last suggestion for future research direction. Detailed study of preceding detected spam reviews to discover spammers' incentives and motivations would be the first step.

Acknowledgment

We would like to thank the anonymous reviewers of this paper for their constructive comments. We would also like to thank Dr.Mohammad Javad Sanjari for his generous consultation to this research project.

References

Ahmed, I., Ali, R., Guan, D., Lee, Y.-K., Lee, S., & Chung, T. C. (2015). Semi-supervised learning using frequent itemset and ensemble learning for SMS classification. *Expert Systems with Applications, 42*(3), 1065–1073.

Akoglu, L., Chandy, R., & Faloutsos, C. (2013). Opinion fraud detection in online reviews by network effects. *ICWSM, 13*, 2–11.

Algur, S. P., Patil, A. P., Hiremath, P. S., & Shivashan, S. (2010). Conceptual level similarity measure based review spam detection. In *Proceedings of 2010 international conference on signal and image processing (ICSIP)* (pp. 416–423). IEEE.

Aye, C. M., & Oo, K. M. (2014). Review spammer detection by using behaviors based scoring methods. In *Proceedings of international conference on advances in engineering and technology*.

Banerjee, S., & Chua, A. Y. K. (2014). Applauses in hotel reviews: Genuine or deceptive? In *Proceedings of science and information conference (SAI)* (pp. 938–942). IEEE.

Blitzer, J., Dredze, M., & Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. *ACL, 7*, 440–447.

Crawford, M., Khoshgoftaar, T. M., Prusa, J. D., Richter, A. N., & Najada, H. Al (2015). Survey of review spam detection using machine learning techniques. *Journal of Big Data, 2*(1), 1–24.

Deng, X., & Chen, R. (2014). Sentiment analysis based online restaurants fake reviews hype detection. *Web Technologies and Applications* (pp. 1–10). Springer International Publishing.

Fayazbakhsh S.K. & Sinha, J. (2012). Review Spam Detection: A Network-based Approach. *Final Project Report: CSE* 590

Feng, S., Banerjee, R., & Choi, Y. (2012). Syntactic stylometry for deception detection. In *Proceedings of the 50th annual meeting of the association for computational linguistics: Short papers: 2* (pp. 171–175).

Fdez-Glez, J., Ruano-Ordas, D., Méndez, J. R., Fdez-Riverola, F., Laza, R., & Pavón, R. (2015). A dynamic model for integrating simple web spam classification techniques. *Expert Systems with Applications, 42*(21), 7969–7978.

Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., & Ghosh, R. (2013). Exploiting burstiness in reviews for review spammer detection. *ICWSM, 13*, 175–184.

Fusilier, D. H., Montes-y-Gómez, M., Rosso, P., & Cabrera, R. G. (2015). Detection of opinion spam with character n-grams. *Computational linguistics and intelligent text processing* (pp. 285–294). Springer International Publishing.

Heydari, A., Tavakoli, M., Ismail, Z., & Salim, N. (2016). Leveraging quality metrics in voting model based thread retrieval. *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, 10*(1), 117–123.

Heydari, A., ali Tavakoli, M., Salim, N., & Heydari, Z. (2015). Detection of review spam: A survey. *Expert Systems with Applications, 42*(7), 3634–3642.

Jindal, N., Liu, B., & Lim, E.-P. (2010). Finding unusual review patterns using unexpected rules. In *Proceedings of the 19th ACM international conference on information and knowledge management* (pp. 1549–1552). ACM.

Jindal, N., & Liu, B. (2008). Opinion spam and analysis. In *Proceedings of the 2008 international conference on web search and data mining* (pp. 219–230). ACM.

Jindal, Nitin, & Liu, Bing (2007). Review spam detection. In *Proceedings of the 16th international conference on World Wide Web* (pp. 1189–1190). ACM.

Kolhe, N. M., Joshi, M. M., Jadhav, A. B., & Abhang, P. D. (2014). Fake reviewer groups' detection system. *Journal of Computer Engineering (IOSR-JCE), 16*(1), 06–09.

Lau, R. Y., Liao, S. Y., Kwok, R. C., Xu, K., Xia, Y., & Li, Y. (2011). Text mining and probabilistic language modeling for online review spam detecting. *ACM Transactions on Management Information Systems, 2*(4), 1–30.

Lim, Ee-Peng, Nguyen, Viet-An, Jindal, Nitin, Liu, Bing, & Lauw, Hady Wirawan (2010). Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM international conference on information and knowledge management* (pp. 939–948). ACM.

Lin, Yuming, Zhu, Tao, Wu, Hao, Zhang, Jingwei, Wang, Xiaoling, & Zhou, Aoying (2014). Towards online anti-opinion spam: Spotting fake reviews from the review sequence. In *Proceedings of 2014 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)* (pp. 261–264). IEEE.

Long, N. H., Nghia, P. H. T., & Vuong, N. M. (2014). Opinion spam recognition method for online reviews using ontological features. *Tạp chí Khoa học*, (61), 44.

Morales, A., Sun, H., & Yan, X. (2013). Synthetic review spamming and defense. In *Proceedings of the 22nd international conference on World Wide Web companion*. International World Wide Web Conferences Steering Committee.

Mukherjee, A., Kumar, A., Liu, B., Wang, J., Hsu, M., Castellanos, M., et al. (2013). Spotting opinion spammers using behavioral footprints. In *Proceedings of the ACM international conference on knowledge discovery and data mining* (pp. 632–640). ACM.

Mukherjee, A., Liu, B., & Glance, N. (2012). Spotting fake reviewer groups in consumer reviews. In *Proceedings of the 21st international conference on World Wide Web*. ACM.

Mukherjee, A., Liu, B., Wang, J., Glance, N., & Jindal, N. (2011). Detecting group review spam. In *Proceedings of the 20th international conference companion on World Wide Web*. ACM.

Ong, T., Mannino, M., & Gregg, D. (2014). Linguistic characteristics of shill reviews. *Electronic Commerce Research and Applications, 13*(2)), 69–78.

Ott, M., Yejin, C., Claire, C., & Jeffrey, T. H. (2011). Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (pp. 309–319). Association for Computational Linguistics.

Ott, M., Claire, C., & Jeff, H. (2012). Estimating the prevalence of deception in online review communities. In *Proceedings of the 21st international conference on World Wide Web* (pp. 201–210). ACM.

Peng, Q., & Zhong, M. (2014). Detecting spam review through sentiment analysis. *Journal of Software, 9.8*, 2065–2072.

Peñalver-Martinez, I., Garcia-Sanchez, F., Valencia-Garcia, R., Rodríguez-García, M. Á., Moreno, V., Fraga, A., et al. (2014). Feature-based opinion mining through ontologies. *Expert Systems with Applications, 41*(13), 5995–6008.

Savage, D., Zhang, X., Yu, X., Chou, P., & Wang, Q. (2015). Detection of opinion spam based on anomalous rating deviation. *Expert Systems with Applications, 42*(22), 8650–8657.

Sharma, K., & Lin, K.-Ip (2013). Review spam detector with rating consistency check. In *Proceedings of the 51st ACM southeast conference* (p. 34). ACM. 2013.

Shashirekha, H. L., Murali, S., & NAgabhushan, P. (2009). Ontology based similarity measure for text documents. In *Proceedings of international conference on signal and image processing (ICSIP)*.

Tavakoli, M., Heydari, A., Ismail, Z., & Salim, N. (2015). A Framework for Review Spam Detection Research. *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, 10*(1), 61–65.

Wang, G., Xie, S., Liu, B., & Yu, P. S. (2011). Review graph based online store review spammer detection. In *Proceedings of 11th international conference on data mining (icdm)* (pp. 1242–1247). IEEE.

Wu, Y., Feng, G., Wang, N., & Liang, H. (2015). Game of information security investment: Impact of attack types and network vulnerability. *Expert Systems with Applications, 42*(15), 6132–6146.

Xie, S., Wang, G., Lin, S., & Yu, P. S. (2012). Review spam detection via temporal pattern discovery. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 823–831). ACM.

Xue, H., Li, F., Seo, H., & Pluretti, R. (2015). Trust-aware review spam detection. In *Trustcom/BigDataSE/ISPA: 1* (pp. 726–733). IEEE.

Ye, Junting, & Akoglu, Leman (2015). Discovering opinion spammer groups by network footprints. *Machine learning and knowledge discovery in databases* (pp. 267–282). Springer International Publishing.