Review Spammer Detection by using Behaviors Based Scoring Methods

Chan Myae Aye, and Kyaw May Oo

***Abstract*—**Product reviews posted at online shopping sites are read by potential customer before deciding to purchase a product. The quality is not control in posting review and trustworthiness of reviews is now a challenging research problem. There is not many published studies on t his topic although web spam and email spam has been investigated extensively. Spammer detection techniques that define spam score based on spamming behaviors of the reviewer are presented in this paper. Review similarity is an important factor to determine spammer. Therefore, all detailed review behaviors of reviewer are calculated if the reviewer writes similar reviews. Rating spam score is also calculated for that reviewer and the more spamming behaviors the reviewers make the more spamming scores they get. The experiments show that the presented technique has comparatively effective spammer detection than other technique based on helpfulness votes alone.

Keywords—spammer detection, review behavior, scoring methods

I. Introduction

THE Web has dramatically changed the way that people express themselves and interact with others. They can now post reviews of products at merchant sites (e.g., amazon.com) and express their views in blogs and forums. It is now well recognized that such *user generated contents* on the Web provide valuable information that can be exploited for many applications. [16] In Web 2.0 application, the user contributed comments offer the promise of a rich source of contextual information about Social Web content. Due to the fact that the quality is not control, anyone can write anything on the Web. This results in many low quality reviews, and worse still *review spam*. This spam review can mislead reader and detection is now a challenging research problem.

Typically, the reviews consist of an overall product score (often in the form of a star-rating) and some free-form review text to allow the reviewer to describe their experience with the product or service in question. Web user can post products reviews at merchant sites to express their views and interact with other users via blogs and forums. Reviewer gives review and also star rating on the product. Figure1 shows the most helpful favorable and the most helpful critical reviews on Amazon web site. It is now well recognized that the user generated content contains valuable information that can be exploited for many applications [4, 13].

Chan Myae Aye, and Kyaw May Oo, University of Computer Studies, Yangon, Myanmar.
(email: cmaye84@gmail.com, kmayoo19@gmail.com)

Fig. 1 Example of reviews on Amazon.com

The existing work has been mainly focused on extracting and summarizing opinions from reviews using natural language processing and data mining techniques [1], [4], [12], [13] and [16]. In the context of Web search, due to the economic and/or publicity value of the rank position of a page returned by a search engine, Web page spam is widespread. [5], [6], [7], [17], [19] and [20] Web page spam refers to the use of "illegitimate means" to boost the rank positions of some target pages in search engines [2], [20]. In the context of reviews, the problem is similar, but also quite different.

Due to the openness of product review sites, spammers can pose as different contributing spammed reviews making them harder to eradicate completely. Spam reviews usually look perfectly normal until one compares them with other reviews of the same products to identify review comments not consistent with the latter. The efforts of additional comparisons by the users make the detection task tedious and non-trivial [11].

Most review spam detection system focus on review behaviors and detect with some classification techniques. Language modeling technique and some similarity computation techniques on review text are also proposed to detect spam and this can have time consuming because of deeply analysis on opinion and text understanding. One should focus on detecting spammers based on their spamming behaviors instead of detecting spam reviews only. Subsequently, spam review can be removed to accelerate the interests of other review users.

The rest of the paper is organized as follows. Section 2 covers some related works. Section 3 presents spammer detection techniques with review based spam score methods.

Experimental evaluation is described in Section 4 and Section 5 is devoted to conclusions.

II. Related Works

Analysis on online opinion becomes a popular research topic recently. Most research trends focus on opinion mining and opinion extraction. A preliminary study of opinion spam was reported in [16].

A spam activities analyzing and spam detection methods are presented in [15]. Three types of spam review such as untruthful opinion, review on brand only and non-review (e.g question and answer and random texts) are also discussed. The analysis said that reviews with negative deviation (in rating) on the same brands give the highest lift curve (that is spam).

The scoring methods to measure the degree of spam for each reviewer is presented in [12] and applies on an Amazon review dataset. They propose target based spamming and deviation based spamming. The results show that the ranking and supervised methods are effective in discovering spammers and outperform the baseline method based on helpfulness votes alone. Only rating deviation and early rating deviation was not much effective.

A language modeling approach for consumer review spam detection is presented in [10]. They showed that Kullback-Leibler (KL) Divergence and the probabilistic language modeling based computational model. This model is effective for the detection of untruthful reviews to estimate the similarity between any pairs of reviews in terms of the likelihood of a review "generating" the contents of another review. Moreover, the Support Vector Machine also called SVM -based method is also effective for the detection of non-reviews. The computational model only detect review spam not detect spammer.

A novel and effective technique, namely, Shingling technique, proposed for detecting spam reviews based on the product features that have been commented in the reviews. A review is considered as a duplicate spam review if its feature matches exactly with the features of the other reviews syntactically. [20] Conceptual level similarity measure used for detecting spam reviews based on the product features is proposed in [21].

Therefore, it can be seen that review similarity is an important factor to detect spammer. In this paper, review similarity is simply calculated by using shingle method to detect suspicious reviewer instead of deeply analysis on opinion mining. Review spam score are calculated based on those similar reviews and rating spam score is also calculated for more accurate results. If a review gets high spam score, reviewer is more likely to be spammer. Spamming behaviors are complicated and cannot easily capture. So that, many researches about review spammer detection are required for improving web sites.

III. Review Behaviors Based Spammer Detection

In this work, spammer detection is targeted on the review behaviors of the reviewer which are given to the products. Most spammers are likely to spam the product with multiple review texts and these review texts are likely to be identical or look similar so as to conserve spamming efforts. So that it is importance to look at how reviews are equal to another review of the same reviewer. When the two reviews are equal, these reviews have more chance to become spam review.

Other review and rating behaviors such as similarity, deviation and good review or bad review and posting date are also considered to detect spammer. The more spamming behaviors the system can detect for a reviewer, the more likely the reviewer is a spammer. Block diagram of behaviors based spammer detection system can be seen in figure 2.

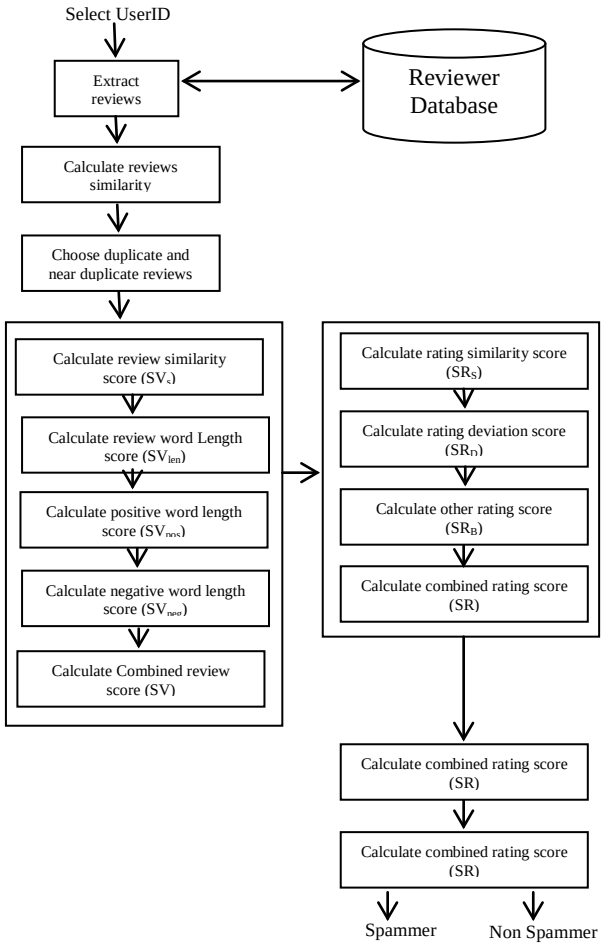

Fig. 2 Block Diagram for Spammer Detection System

The system consists of three portions: 1. Review Based Spam Score, 2. Rating Based Spam Score and 3. Combined scores. Before these three steps, it is needed to calculate review similarity for all reviews of reviewer u_i. Then choose duplicate review if similarity score is equal 1 and near duplicate review if similarity value is between 0.99 and 0.75. Review based spam score is calculated for this duplicate and near duplicate reviews.

TABLE I
Defination and Notations

U	Set of users
$\mathbf{V_{sim}}$	Set of duplicate and near duplicate reviews
$\mathbf{V_{i*}}$	Set of reviews by user u_i to all products
$\mathbf{E_{i*}}$	Set of rating by user u_i to all products

The spammer detection algorithms called calculateSpamScore() and ReviewSpamScore() and RatingSpamScore() are described in figure 3,4 and 5. The next subsection describes all proposed spam score methods.

Calculate Spam Score( )
Input: Reviewer u_i from set of reviewer U
Initial: **simthreshold, spamthreshold**
For each reviewer u_i in U
1: Extract all reviews of reviewer u_i
$\mathbf{V_{i*}=\{v_1,v_2,\ldots.v_j\}}$
2: Calculate review similarity for all reviews in $\mathbf{V_{i*}}$
For k=1 to j
For l=k+1 to j
2.1: Calculate review similarity
$sim(v_k, v_l) = \frac{s(v_k w) \cap s(v_l w)}{s(v_k w) \cup s(v_l w)}$
.2: If sim(v_k,v_l) > **simthreshold** then,
Set similarity value v_k,v_l to V_{sim}
End for
End for
Step 3: Calculate review similarity spam score
$\bar{SV_S}(u_i) = avg_{v_k \in V_{sim}}\ sim\ (v_k)$
Step 4: Call ReviewSpamScore(V_{sim},SV_s (u_i))
Step 5: Call RatingSpamScore(u_i)
Step 6: Calculate all Spam score
$CRS(u_i) = \frac{1}{2}[SV(u_i) + SR(u_i)]$
Step 7: Normalized score
$CS(u_i) = \frac{CSR_{u_t}}{max_{u_{t'} \in U_t} CSR_{u_{t'}}}$
Step 8: If CS(u_i) > **spamthreshold**
Label Reviewer u_i as spam
Else label Reviewer u_i as nonspam
End if
End

Fig. 3 CalculateSpamScore() Algorithm

ReviewSpamScore(V_{sim},SV_s (u_i))
For all reviews in V_{sim}
Step 1: Calculate Review Word Length Score
$\bar{pro}(v_k) = \frac{n(v_k)}{\max_{u_i \in U} n(v_k)}$
Step 2: Calculate Number of Positive Word Length Score
$\bar{pos}(v_k) = \frac{npos(v_k)}{n(v_k)}$
Step 3: Calculate Number of Negative Word Length Score
$\bar{neg}(v_k) = \frac{nneg(v_k)}{n(v_k)}$
Step 4: Calculate Other Review Score
$orv(v_k) = \frac{1}{2}\ [pos(v_k) + fpos(v_k)]$
$pos(v_k) = \begin{cases} 1, if\ review\ v\ is\ the\ first\ five\ review \\ 0, else \end{cases}$
$fpos(v_k) = \begin{cases} 1, if\ review\ v\ is\ the\ first\ review \\ 0, else \end{cases}$
Step 5: Calculate average Review Score
$SV_{len}(u_i) = avg\ pro(v_k)$
$SV_{pos}(u_i) = avg\ pos(v_k)$
$SV_{neg}(u_i) = avg\ neg(v_k)$
$SV_B(u_i) = avg\ orb(v_k)$
Step 6: Calculate Combined Review Score
$\bar{SV}(u_i) = \frac{1}{3}\ SV_S(u_i) + \frac{1}{6}\ SV_{len}(u_i) + \frac{1}{6}\ SV_{pos}(u_i) + \frac{1}{6}\ SV_{neg}(u_i) + \frac{1}{6} SV_B(u_i)$
Return SV**(u_i)**
End

Fig. 4 ReviewSpamScore() Algorithm

RatingSpamScore(U_i)
For all rating of reviewer u_i
Step 1: Extract all rating of reviewer u_i in E_{i*}
$\mathbf{E_{i*}=\{e_1,e_2,\ldots.e_j\}}$
For all rating in E_{i*}
2.1: Calculate Rating Similarity Score
$\text{sim}(r_i, e_k) = 1 - |\bar{r_i} - e_k|$
2.2: Calculate Rating Deviation Score
$dev(e_k) = e_k - Avg_{e_k \in E_{i*} \wedge e_{k'} \in E_{*j}}\ e_{k'}$
2.3: Other Rating Behaviors Score
$orb(e_k) = \frac{1}{4}\ [r(e_k) + rgb(e_k) + rbg(e_k) + ndev(e_k))]$
$r(e_k) = \begin{cases} 1, if\ e\ is\ good\ rating\ or\ bad\ rating \\ 0, else \end{cases}$
$rgb(e_k) = \begin{cases} 1, if\ rating\ e\ is\ bad\ rating\ and\ come \\ \quad after\ the\ first\ bad\ rating \\ 0, else \end{cases}$
$rbg(e_k) = \begin{cases} 1, if\ rating\ e\ is\ good\ rating\ and\ come \\ \quad after\ the\ first\ good\ rating \\ 0, else \end{cases}$
$ndev(e_k) = \begin{cases} 1, if\ dev(e) is\ negative \\ 0, else \end{cases}$
End for
Step 3: Calculate average Rating Score
$SR_S(u_i) = \frac{\max_{\forall r_i \in R} sim(R)}{|E_{i*}|}$
$SR_D(u_i) = avg\ |dev(e_k)|$
$SR_B(u_i) = Avg_{e \in e_{i*}} orb(e_k)$
Step 4: Combined Score
$SR(u_i) = \frac{1}{2}\ SR_S(u_i) + \frac{1}{4} SR_D(u_i) + \frac{1}{4}\ SR_B(u_i)$
Step 5: Return SR(u_i)
End

Fig. 5 RatingSpamScore() Algorithm

A. *Review Similarity Spam Score ($SV_S(u_i)$)*

As described above, a reviewer spam a product with multiple review texts and such reviews texts are likely to be identical or look similar. Cosine similarity with bag of words is used in similarity computation of review but TFIDF of cosine has less similarity for rare terms. So, the system applies shingle method to measure reviews similarity.

To get review similarity score the following equation is used.

$$SV_S(u_i) = avg_{v_k \in V_{sim}}\ sim\ (v_k) \quad (1)$$

where, sim(v_k) is similarity value for review v_k in V_{sim} . V_{sim} is a collection of duplicate and nearly duplicate reviews for reviewer u_i.

If similarity scores are high, the spamming scores are also high.

B. *Review word length score ($pro(v_k)$)*

The length of the review is also an indication to detect spam. Most spammer writes long review to get the reviewer attention. The review word length score can be obtained by dividing number of words in review n(v_k) by maximum number of words of other review given by all reviewer.

$$pro(v_k) = \frac{n(v_k)}{\max_{u_i \in U} n(v_k)} \quad (2)$$

C. *Positive and Negative word length score ($pos(v_k)$, $neg(v_k)$)*

Spammer also used more positive words or negatives words to express their opinion that the product is good or bad. So that some opinion bearing words in the review such as

"beautiful", "great", "bad" and "poor" can be consider for spam detection.

The positive or negative word length score can be obtained by dividing number of positive words npos(v_k) or negative words npos(v_k) in review by number of words in review. The equations for positive and negative word length score can be seen The equations for positive and negative word length score are:

$$pos(v_k)=\frac{npos(v_k)}{n(v_k)} \quad (3)$$

$$neg(v_k)=\frac{nneg(v_k)}{n(v_k)} \quad (4)$$

D. Other Review Spam Score ($orv(v_k)$)

Only review similarity is not a good indicator to detect spammer Therefore the system is considered on the other review behaviors that likely to be spammer. These behaviors are :

(1) Reviews which are written early tend to get more reviewer attention, and thus can have bigger impact on the sale of a product. The position of the review date is also an important factor because early reviews are more concentrated by users.

(2) The first reviews are more likely to be spam than other reviews. Some spammer tends to write review as soon as the product release because first review is very important for new product and first reviews are more concentrated by users.

So the system looks on these behaviors to get accurate spam score. The more suspicious behaviors the reviewer act the more spam score the reviewer get.

$$SV_B(u_i) = avg\ orv(v_k) \quad (5)$$

where, orv(v_k) is other review behaviors of review v_k [see in figure. 4)

E. Combined Review Spam Score ($SV(u_i)$)

Combined review spam score is the spam score of reviewer to the products by all of their review spamming behaviors. It achieves effective spam score by simply giving some weight of each score function. In this case, review similarity score get more weight because the similar review behavior is more likely to be spammer than other behaviors of the reviewer.

$$SV(u_i)=\frac{1}{3}\ SV_S(u_i)+\frac{1}{6}\ SV_{len}(u_i)+\frac{1}{6}\ SV_{pos}(u_i)\ +\frac{1}{6}\ SV_{neg}(u_i)\ +\frac{1}{6}\ SV_B(u_i) \quad (6)$$

F. Rating Similarity Score (SR_s)

Rating similarity is an important factor to determine spammer as like review similarity. Spammer gives same rating to the product to conserve their spamming effort. The rating similarity score can be obtained by using the following equation:

$$SR_S(u_i) = avg\ sim(e_k, e_{k+1}) \quad (7)$$

where, sim(e_k,e_{k+1}) is similarity between two rating. (See in RatingSpamScor())

G. Rating Deviation Score ($SR_D(u_i)$)

Reasonable reviewer is expected to give ratings that are similar to other rating of user on the same product. Mostly, the ratings of the spammer are different from these reasonable ratings. Spammer tends to give high rating in low quality product to promote that product and also give low rating in high quality product to damage that product reputation. The spammer's rating can be deviated from other product rating. The rating deviation score can be obtained by the following equation.

$$SR_D(u_i)= avg\ |dev(e_k)| \quad (8)$$

where, dev(e_k) is deviation of rating e_k. (See in Figure. 5)

H. Other Rating Score ($SR_B(u_i)$)

Only rating similarity and deviation is not a good indicator to obtain effective detection, so the system is considered on other ratings behaviors that likely to be spammer. These behaviors are

(1) Some spammers give very high rating and very low rating to promote or demote product.

(2) Spammer writes good rating just after bad rating and also writes bad rating just after good rating to damage control.

(3) Reviewers are biased towards a brand and give reviews with negative deviation on products of that brand are very likely to be spammer.

So, all these behaviors are considered to detect spammer. Other rating behaviors can be obtained by the following equation.

$$SR_B(u_i) = Avg\ orb(e_k) \quad (9)$$

where, orb(e_k) is other rating behaviors of review (See in Figure. 5)

I. Combined Rating Spam Score ($SR(u_i)$)

Combined spam score is the spam score of reviewer to the products by their rating spamming behaviors. In this case, rating similarity score get more weight because similar rating is more likely to be spammer than others.

$$SR(u_i)=\frac{1}{2}\ SR_S(u_i)+\frac{1}{4}SR_D(u_i)+\frac{1}{4}\ SR_B(u_i) \quad (10)$$

J. Combined Spam Score ($CS(u_i)$)

Finally, review spam score and rating spam score are needed to combine and the effective spam score can be obtained by the following equation.

$$CRS(u_i) = \frac{1}{2}[SV(u_i) + SR(u_i)] \quad (11)$$

Then the spam score is normalized by (12) and if score is greater than threshold value, the system defines that reviewer as spammer. If not, defines that reviewer as nonspammer.

$$CS(u_i) = \frac{CSR_{u_i}}{max_{u_{i'} \in U_i} CSR_{u_{i'}}} \quad (12)$$

IV. Evaluation

The presented methods are evaluated by using data from amazon.com. The reason for using this data set (http://131.193.40.52/data) is that it is large and covers a very wide range of products. Amazon is considered one of the most successful e-commerce Web sites with a relatively long history. This dataset gives the information such as Product ID, Reviewer ID, Rating, Date, Review Title, Review Body, Number of Helpful Feedbacks and Number of feedbacks. The statistics of the dataset are presented in table (2).

TABLE II
Dataset Statistics

	Number
Number of Reviewers	1,037,621
Number of Product	962,234
Number of reviews	3,794,694

There is still no gold standard dataset to test the accuracy and this is why detection of review spam has been neglected so far. Therefore, the system uses three human evaluators to check whether a reviewer is spammer or not. Human evaluation has also been used for opinion spam in prior works [12]. It is however important to note that just by reading a single review without any context; it is very hard to determine whether a review is fake (spam) or not [14]. However, it has been shown in [1] that when sufficient context is provided e.g., reviewing patterns, ratings, type/brand of products reviewed, posting activity trails, etc., human expert evaluation becomes easier. The evaluator can look at the feature of each reviewer's review behaviors and can give feedback whether the reviewer is spammer or not.

There are many reviewers and it is impossible for human evaluators to judge everyone that can be very time consuming (see Table 2). Only small subsets of reviewers are evaluated to handle this issue. All reviewers are ranked from higher to lower score for each evaluation method. Select top 10 reviewers (spammer) and bottom 10 reviewers (nonspammer) from those ranks and then merge all the selected spammers into a pool which consists of 64 reviewers. These reviewers are then sorted by their combined spamming behavior scores. 25 top ranked reviewers and 25 bottom ranked reviewers are then selected for user evaluation. This number of reviewers is quite reasonable for a human evaluator to examine. The system further randomly orders the reviewers so that there is no relationship between reviewers' order and their spammer scores.

To judge each reviewer is spammer or not, select his/her reviews to be highlighted for human evaluator's attention. The reviews are selected based on their involvement in the spamming behaviors identified. Specifically, identical (or similar) review with other reviews position of review date, the only review or not and rating behaviors like deviation, similarity and good or bad rating.

Three human evaluators are recruited to examine the selected reviewers and rating. For each reviewer, the labeled decision is either spammer" and non-spammer" and the evaluators are not informed about the number of spammers to be labeled. Given the results of the three evaluators, final label to each reviewer is assigned by using majority voting. A reviewer is assigned a final spam or non-spam label if the label is agreed by two or more evaluators.

V. Results

Table 3 shows the number of spammers and non-spammer labeled by the evaluators in the diagonal cell and the number of overlapping spammers between each pair of evaluators. The evaluators are largely consistent in their judgments of spammers and non-spammers. All the three evaluators agree on 19 spammers and 20 non-spammers. Given the results of the three evaluators, final label to each reviewer is assigned by using majority voting. A reviewer is assigned a final spam or non-spam label if the label is agreed by two or more evaluators. The system ending up having 23 reviewers as spammer (17 reviewer have 3 votes and 6 reviewer has 2 votes each) 27 reviewers as non-spammer (9 reviewers have 1 vote and 18 reviewers have 0 vote) Cohen's kappa values of the evaluator pairs are between 0.64 and 0.84**.** It has substantial agreement because the kappa value can be interpreted as:

- No agreement (for k<0)
- Slight agreement (for Εk(0,0.2])
- Fair agreement (for Εk(0.2,0.4])
- Moderate agreement (for Εk(0.4,0.6])
- Substantial agreement (for Εk(0.6,0.8])
- Almost perfect agreement (for Εk(0.8,1s])

The number of top 10 reviewers with the final spammer labels, and the number of bottom 10 reviewers with non-spammer labels for different methods are shown in Table 4. Rating based spam score is a base line method and the presented techniques are significantly better than base line method based on helpfulness vote.

TABLE III
The Result Of Evaluation by Three Human Evaluators

	Evaluator 1	Evaluator 2	Evaluator 3
# Spammers			
Evaluator 1	**27**	**21**	**20**
Evaluator 2		**24**	**21**
Evaluator 3			**22**
# Non-spammers			
Evaluator 1	**23**	**20**	**21**
Evaluator 2		**26**	**25**
Evaluator 3			**28**

TABLE IV
Top 10 spammers and Bottom 10 non Spammers

	Spam score methods			
	Based Line	**Review Based spam score**	**Rating based spam score**	**Combined spam score**
#Spammer in top 10	7	10	9	10
#nonspammer in bottom 10	7	10	9	10

a. System Accuracy

Text Retrieval Conferences (TREC) spam track (Tormack and Lynam. 2005; Cormack 2007) is used to measure the performance of the presented techniques. They were widely used to evaluate other kinds of Web spam. With reference to a confusion matrix depicted in table [5], the various effectiveness measures can be defined by:

$$hm = \frac{b}{b+d} \quad (13)$$

$$sm = \frac{c}{a+c} \quad (14)$$

$$lam = logit^{-1}\left(\frac{logit(hm)+logit(sm)}{2}\right) \quad (15)$$

$$tp = \frac{a}{a+C} \quad (16)$$

where a, b, c, and d refer to the number of reviews falling into each category presented in table [5]. The ham misclassification rate (*hm*) is the fraction of all ham misclassified as spam; the spam misclassification rate (*sm*) is the fraction of all spam misclassified as ham. It is desirable to have a single measure which combines both of the above measures. Therefore, the TREC Spam track also made use of the logistic average misclassification rate (*lam*) to measure the effectiveness of spam detection systems, where $logit^{-1}(x) = \frac{e^x}{1+e^x}$ and $logit(x) = \ln\left(\frac{x}{1-x}\right)$.

Since *hm*, *sm* and *lam* are the measures for failure rather than effectiveness, the lower scores imply a better detection performance. The true positive rate (*tp*) is the fraction all spam identified by the system. On the other hand, the common effectiveness measure accuracy=$\frac{a+d}{a+b+c+d}$ may be measured for spam detection System. The accuracy figure is report in table [6]. These values are calculated from results of the user evaluation tests on 50 reviewers. The presented techniques have more accuracy than base line method.

TABLE V
CONFUSION MATRIX FOR THE DEFINITION OF THE EFFECTIVENESS MEASURE

System's Classification	Gold Standard – Human Classification		
		Spam	**Ham**
	Spam	**a**	**b**
	Ham	**c**	**d**

TABLE VI
COMPARATIVE PERFORMANCE OF SPAMMER DETECTION TECHNIQUES

	Spam score methods			
	Baseline	**Review based spam score**	**Rating based Spam Score**	**Combined spam score**
hm%	0.30	0.00	0.10	0.00
sm%	0.30	0.00	0.10	0.00
lam%	0.65	0.00	0.10	0.00
tp%	0.70	1	0.90	1
Accuracy	0.70	1	0.90	1

VI. CONCLUSION AND FUTURE WORK

This paper presents review spammer detection techniques based on reviewer spamming behaviors. Spam scores for each reviewer are high if they do spamming behaviors. The system proposed review based spam score and his score is also combine with rating based spam score and the presented techniques have better results than baseline method. The techniques focus only on scoring methods and has not considered about deeply understanding of review text. So it is needed to add some classification method to achieve more effective spammer detection system from this result.

REFERENCES

[1] A. Mukherjee, B. Liu, and N. Glance, 2012. "Spotting Fake Reviewer Groups in Consumer Reviews," WWW (2012)

[2] A. M. Popescu and O. Etzioni. "Extracting Product Features and Opinions from Reviews," *EMNLP'2005*.

[3] A. Ntoulas, M. Najork, M. Manasse & D. Fetterly. "Detecting Spam Web Pages through Content Analysis," WWW'2006.

[4] B. Liu. Web Data Mining: Exploring hyperlinks, contents and usage data, Springer, 2007.

[5] B. Pang, L. Lee & S. Vaithyanathan. "Thumbs up? Sentiment classification using machine learning techniques," *EMNLP'2002*.

[6] B. Wu and B. D. Davison. "Identifying link farm spam pages," WWW'06, 2006.

[7] B. Wu, V. Goel & B. D. Davison. "Topical TrustRank: using topicality to combat Web spam," *WWW'2006*.

[8] C. Castillo, D. Donato, L. Becchetti, P. Boldi, S. Leonardi, M. Santini, S. Vigna. "A reference collection for web spam," SIGIR Forum'06, 2006.

[9] C. Danescu-Niculescu-Mizil, G. Kossinets, J. Kleinberg, and L. Lee. "How opinions are received by online communities: a case study on amazon.com helpfulness votes," In WWW, 2009.

[10] C.L. Lai, K.Q. Xu, Raymond Y.K. Lau ,Y. Li and L. Jing, "A language modeling approach for consumer review spam detection," IEEE International Conference on E-Business Engineering , 2010.

[11] K. Dave, S. Lawrence & D. Pennock. "Mining the peanut gallery: opinion extraction and semantic classification of product reviews," *WWW'2003*.

[12] L . Ee-Peng, V. Nguyen, N. Jindal, B. Liu, H. W. Lauw. "Detecting Product Review Spammers using Rating Behaviors," CIKM'10, Toronto, Ontario, Canada, October 26–30, 2010.

[13] M. Hu & B. Liu. "Mining and summarizing customer reviews," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-2004, full paper)*, Seattle, Washington, USA, Aug 22-25, 2004.

[14] M. Ott, , Y. Choi, , C. Cardie, and J.T. Hancock, 2011. "Finding Deceptive Opinion Spam by Any Stretch of the Imagination." ACL, 309–319 (2011).

[15] N. Jindal and B. Liu. "Opinion Spam and Analysis." WSDM'08, Palo Alto, California, USA, February 11-12, 2008.

[16] N. Jindal and B. Liu. "Review Spam Detection" WWW 2007, Poster Paper, Banff, Alberta, Canada, May 8–12, 2007.

[17] P.Turney. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. *ACL'2002*.

[18] 17.R. Baeza-Yates, C. Castillo & V. Lopez. PageRank increase under different collusion topologies. *AIRWeb'05*,

[19] S. P. Algur, A.P. Patil, P.S Hiremath, S. Shivashankar, "Conceptual level Similarity measure based review spam detection" International Conference on Signal and Image Processing, IEEE, 2010.

[20] S. P. Algur, A.P. Patil, P.S Hiremath, S. Shivashankar, "Spam Detection of Customer Reviews from Web Pages,"

[21] Y. Wang, M. Ma, Y. Niu, H. Chen. "Spam Double-Funnel: Connecting Web Spammers with Advertisers," *WWW2007*.

[22] Z. Gyongyi & H. Garcia-Molina. "*Web Spam Taxonomy*," Technical Report, Stanford University, 2004.