Spam Review Detection and Recommendation of Superior Results in NetSpam Framework

KALYANI PATIL, RUTUJA FAKIRE, PALLAVI AGWANE, ARTI RODE
BE Students, Dept. of IT, MMCOE Pune.

PRANJALI KUCHE
Professor, Dept. of IT, MMCOE Pune.

Abstract- Major Society of people using internet trust the contents of net. The liability that anyone can take off a survey give a brilliant chance to spammers to compose spam surveys about hotels and services for various interests. Recognizing these spammers and the spam content is a widely debated issue of research and in spite of the fact that an impressive number of studies have been done as of late towards this end, yet so far the procedures set forth still scarcely distinguish spam reviews, and none of them demonstrate the significance of each extracted feature type. In this application, use a novel structure, named NetSpam, which proposes spam features for demonstrating hotel review datasets as heterogeneous information networks to design spam review detection method into a classification issue in such networks. Utilizing the significance of spam features helps us to acquire better outcomes regarding different metrics on review datasets. The outcomes represent that NetSpam results with the previous methods and encompassed by four categories of features; involving review-behavioral, user-behavioral, review linguistic, user-linguistic, the first type of features performs better than the other categories. The contribution work is when user will search query it will display all top products as well as there is recommendation of the product.

Keywords- Social Media, Social Network, Spammer, Spam Review, Fake Review, Heterogeneous Information Networks, Sentiment Analysis, Semantic Analysis

I. INTRODUCTION

Social Media portals play an important role in information propagation. Today a lot of people rely on the written reviews of other users in the selection of products and services. Additionally written reviews help service providers to improve the quality of their products and services. The reviews therefore play an important role in success of a business. While positive reviews can provide boost to a business, negative reviews can highly affect credibility and cause economic losses. Since anyone can leave comments as review, provides a tempting opportunity for spammers to write spam reviews which mislead users' choices. A lot of techniques have been used to identify spam reviews based on linguistic patterns, behavioral patterns. Graph based algorithms are also used to identify spammers. However many aspects are still unsolved. The general concept of NetSpam framework is to build a retrieved review dataset as a Heterogeneous

Information Network (HIN) and to convert the problem of spam detection into a classification problem. In particular, convert review dataset as a HIN in which reviews are connected through different features. A weighting algorithm is then employed to calculate each feature's importance. These weights are then used to calculate the very last labels for reviews using both unsupervised and semi-supervised procedures.

NetSpam is able to find features' importance relying on metapath definition and based on values calculated for each review. NetSpam improves the accuracy and reduces time complexity. It highly depends to the number of features used to identify spam reviews. Thus using features with more weights will resulted in detecting spam reviews easier with lesser time complexity.

II. LITERATURE SURVEY

The pair wise features are first explicitly utilized to detect group colluders in online product review spam campaigns, which can reveal collusions in spam campaigns from a more fine-grained perspective. A novel detecting framework [1] named Fraud Informer is proposed to cooperate with the pair wise features which are intuitive and unsupervised. Advantages are: Pair wise features can be more robust model for correlating colluders to manipulate perceived reputations of the targets for their best interests to rank all the reviewers in the website globally so that top-ranked ones are more likely to be colluders. Disadvantage is difficult problem to automate.

The paper [2] proposes to build a network of reviewers appearing in different bursts and model reviewers and their co-occurrence in bursts as a Markov Random Field (MRF) and apply the Loopy Belief Propagation (LBP) method to induce whether a reviewer is a spammer or not in the graph. A novel assessment method to evaluate the detected spammers automatically using supervised classification of their reviews. Advantages are: High accuracy, the proposed method is effective. To detect review spammers in review bursts. To detect spammers automatically. Disadvantage is: a generic framework is not used for detect spammers.

In [3] paper, the challenges are: The detection of fraudulent behaviors, determining the trustworthiness of review sites, since some may have strategies that enable misbehavior, and creating effective review aggregation solutions. The TrueView score, in three different variants, as a proof of concept that the synthesis of multi-site views can provide important and usable information to the end user. Advantages are: develop novel features capable of finding cross-site discrepancies effectively, a hotel identity-matching method with 93% accuracy. Enable the site owner to detect misbehaving hotels. Enable the end user to trusted reviews. Disadvantage is difficult problem to automate.

In [4] paper describes unsupervised anomaly detection techniques over user behavior to distinguish probably bad behavior from normal behavior. To find diverse attacker schemes fake, compromised, and colluding Facebook identities with no a priori labeling while maintaining low false positive rates. Anomaly detection technique to

forcefully identify anomalous likes on Facebook ads. Achieves a detection rate of over 66% (covering more than 94% of misbehavior) with less than 0.3% false positives. The attacker is trying to drain the budget of some advertiser by clicking on ads of that advertiser.

In [5] paper, a grouped classification algorithm called Multi-typed Heterogeneous Collective Classification (MHCC) and then extends it to Collective Positive and Unlabeled learning (CPU).The proposed models can markedly increase the F1 scores of strong baselines in both PU and non-PU learning environment. Advantages are: Proposed models can markedly increase the F1 scores of strong baselines in both PU and non-PU learning settings. Models only use language self-contained features; they can be smoothly generalized to other languages. Identifies a large number of implied fake reviews hidden in the unlabeled set. Fake reviews hiding in the unlabeled reviews that Dianping's algorithm did not capture. The ad-hoc labels of users and IPs used in MHCC may not be very specific as they are computed from labels of neighboring reviews.

III. OPEN ISSUES

Online Social Media websites play a main role in information propagation which is considered as an important source for producers in their advertising operations as well as for customers in selecting products and services. People mostly believe on the written reviews in their decision-making processes, and positive/negative reviews encouraging/discouraging them in their selection of products and services. These reviews that reason have emerge as an important issue in fulfillment of a business even as positive opinions can carry blessings for an employer, bad evaluations can probably effect credibility and motive monetary losses. The critiques written to change customers' perception of ways top a product or a service are taken into consideration as spam, and are regularly written in trade for money.

Disadvantages:

1) There is no information filtering concept in online social network.

2) People believe on the written reviews in their decision-making processes, and positive/negative reviews encouraging/discouraging them in their selection of products and services.

3) Anyone create registration and gives comments as reviews for spammers to write fake reviews designed to misguide user's opinion.

4) Less accuracy.

5) More time complexity.

IV. SYSTEM OVERVIEW

A novel proposed framework is to representative a given review dataset as a Heterogeneous Information Network (HIN) and to solve the issue of spam detection into a HIN classification issue. In particular, to show the review dataset as a HIN in which reviews are connected through different node types (such as features and users). A weighting algorithm is then employed to calculate each features importance (or weight). These weights are applied to calculate the final labels for reviews using both unsupervised and supervised methods. Based on our observations, defining two views for features (review-user and behavioral-linguistic), the classified features as review behavioral have more weights and yield better performance on spotting

spam reviews in both semi-supervised and unsupervised approaches. The feature weights can be added or removed for labeling and hence time complexity can be scaled for a specific level of accuracy. Categorizing features in four major categories (review-behavioral, user-behavioral, review-linguistic, user-linguistic), helps us to understand how much each category of features is contributed to spam detection.

- NetSpam framework that is a novel network based approach which models review networks as heterogeneous information networks.
- A new weighting method for spam features is proposed to determine the relative importance of each feature and shows how effective each of features are in identifying spams from normal reviews.
- NetSpam framework increases the accuracy as opposed to the state-of-the art in terms of time complexity, which distinctly relies upon to the variety of capabilities used to perceive an unsolicited spam evaluation.

A. Architecture

The Fig.1 shows the proposed system architecture. The general concept of our proposed framework is to model a given review dataset as a Heterogeneous Information Network and to map the problem of spam detection into a HIN classification problem. In particular, model review dataset as in which reviews are connected through different node types.

A weighting algorithm is then employed to calculate each features importance. These weights are applied to calculate the final labels for reviews using both unsupervised and supervised techniques. This is based on the observations defining two views for features.

Advantages of Proposed System:

1) It identifies spam and spammers as well as different type of analysis on this topic.

2) Written reviews also help service providers to enhance the quality of their products and services.

3) It identifies the spam user using positive and negative reviews in online social media.

4) It displays only trusted reviews to the users.

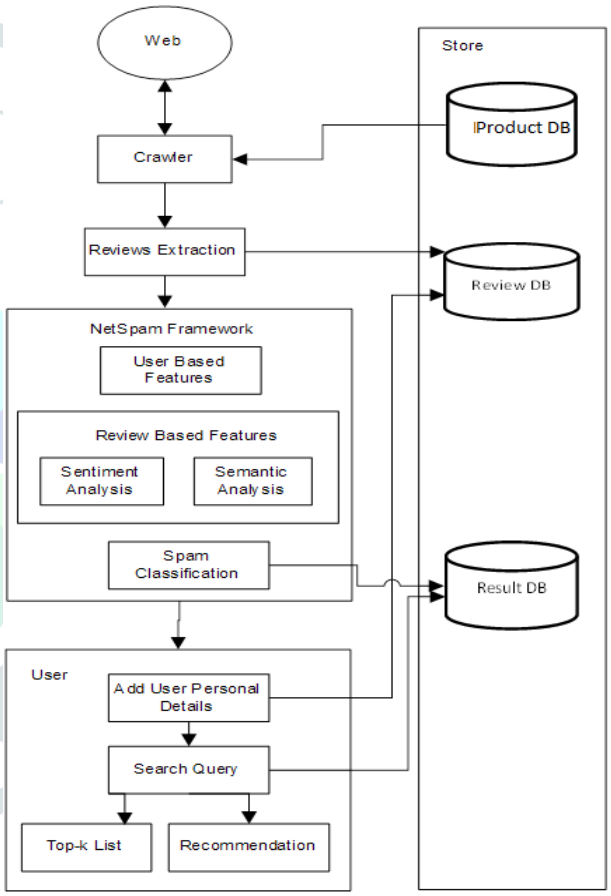

Fig. 1 System Architecture

1. Sentiment Analysis using Sentiwordnet Dictionary

```
polarizedTokensList ← newList()
while tokenizedTicket.hasNext() do
        token←tokenizedTicket.next()
        lemma←token.lemma
        polarityScore←null
        if DomainDictionary.contains(lemma,pos)
then
```

```
        if
SentiWordNet.contains(lemma,pos) and
SentiWordNet.getPolarity(lemma,pos) != 0)
then
            polarityScore ← SentiWordNet.getPolarity(lemma, pos)
        else
            domainDicToken←DomainDictionary.getToken(lemma, pos)
            if domainDicToken.PolarityOrientation == ”POSITIVE” then
                polarityScore ← DefaultPolarity.positive
            else
                polarityScore ← DefaultPolarity.negative
            end if
        end if
        polarizedTokensList.add(token, polarityScore)
    end if
end while
return polarizedTokensList
```

2. NetSpam Algorithm:

Input: review_dataset, spam_feature_list, pre_labeled_reviews

Output: features_importance (W), spamicity_probability (Pr)

Step 1: u, v: review, y_u: spamicity probability of review u

Step 2: $f(x_{lu})$: initial probability of review u being spam

Step 3: P_l metapath based on feature l, L: features number

Step 4: n: number of reviews connected to a review

Step 5: m_u^{Pl} : the level of spam certainty

Step 5: $m_{u,v}^{Pl}$: the metapath value

Step 6: Prior Knowledge

Step 7: **if** semi-supervised mode

Step 8: **if** $u \in pre\_labeled\_reviews$

Step 9: $y_u = label(u)$

Step 10: **else**

Step 11: $y_u = 0$

Step 12: **else** unsupervised mode

Step 13: $y_u = \frac{1}{L}\sum_{l=1}^{L} f(x_{lu})$

Step 14: Network Schema Definition

Step 15: schema = defining schema based on spam-feature-list

Step 16: Metapath Definition and Creation

Step 17: **for** $P_l \in schema$

Step 18: **for** $u, v \in review\_dataset$

Step 19: $m_u^{Pl} = \frac{\lfloor s \times f(x_{lu}) \rfloor}{s}$

Step 20: $m_v^{Pl} = \frac{\lfloor s \times f(x_{lv}) \rfloor}{s}$

Step 21: **if** $m_u^{Pl} = m_v^{Pl}$

Step 22: $m_{u,v}^{Pl} = m_u^{Pl}$

Step 23: **else**

Step 24: $m_{u,v}^{Pl} = 0$

Step 25: Classification - Weight Calculation

Step 26: **for** $Pl \in schemes$

Step 27: **do** $W_{Pl} = \frac{\sum_{r=1}^{n}\sum_{s=1}^{n} mp_{r,s}^{Pl} \times y_r \times y_s}{\sum_{r=1}^{n}\sum_{s=1}^{n} mp_{r,s}^{Pl}}$

Step 28: Classification - Labeling

Step 29: **for** $u, v \in review\_dataset$

Step 30 $Pr_{u,v} = 1 - \prod_{Pl=1}^{L} 1 - m_{u,v}^{Pl} \times W_{Pl}$

Step 31: $Pr_u = avg(Pr_{u,1}, Pr_{u,2}, \dots, Pr_{u,n}$

Step 32: return (W, Pr)

3. Algorithm Top-K-Join-Tuple (R, S, j, K, T)

Input: relation R, relation S, the rank function f, the number of join tuples K, and the lower bound T of the rank function;

Output: top-K tuples from R that can be joined with tuples from S,

Process:

Begin

k:=0; //Number of tuples in R that has a join candidate in S

u:=0; //Row number of the current tuple in S

While k<K and u< S.length

u: =u+ 1 ;

v:=0; // Row number of the current tuple in R

While k<K and v<R.Iength

v:=v+1;

If tuple S (u) and tuple R (v) satisfy the join condition and

f(R (v).r (p), S (u). S(q)) is greater than T

Then

Output (v, u, f) to the rank queue of R;

k:=k+l;

End If

End While

End While

End

C. ***Spam Features:***

User-Behavioral (UB) based features:

Burstiness: Spammers, usually write their spam reviews in short period of time for two reasons: first, because they want to impact readers and other users, and second because they are temporal users, they have to write as much as reviews they can in short time.

$$x_{BST}(i) = \begin{cases} 0 & (L_i - F_i) \notin (0, \tau) \\ 1 - \frac{L_t - F_t}{\tau} & (L_i - F_i) \in (0, \tau) \end{cases} \quad (1)$$

Where,

$L_i - F_i$ describes days between last and first review for τ = 28.

Users with calculated value greater than 0.5 take value 1 and others take 0.

User-Linguistic (UL) based features:

Average Content Similarity, Maximum Content Similarity: Spammers, often write their reviews with same template and they prefer not to waste their time to write an original review. In result, they have similar reviews. Users have close calculated values take same values (in [0; 1]).

Review-Behavioral (RB) based features:

- Early Time Frame: Spammers try to write their reviews a.s.a.p., in order to keep their review in the top reviews which other users visit them sooner.

$$x_{ETF}(i) = \begin{cases} 0 & (L_i - F_i) \notin (0, \delta) \\ 1 - \frac{L_t - F_t}{\delta} & (L_i - F_i) \in (0, \delta) \end{cases} \quad (2)$$

Where,

$L_i - F_i$ denotes days specified written review and first written review for a specific business. We have also δ = 7. Users with calculated value greater than 0.5 takes value 1 and others take 0.

- Rate Deviation using threshold: Spammers, also tend to promote businesses they have contract with, so they rate these businesses with high scores. In result, there is high diversity in their given scores to different businesses which is the reason they have high variance and deviation.

$$x_{DEV}(i) = \begin{cases} 0 & Otherwise \\ 1 - \frac{r_{ij} - avg_{e \in E*j} r(e)}{4} > \beta_1 \end{cases} \quad (3)$$

Where,

β_1 is some threshold determined by recursive minimal entropy partitioning. Reviews are close to each other based on their calculated value, take same values (in [0; 1)).

Review-Linguistic (RL) based features:

Number of first Person Pronouns, Ratio of Exclamation Sentences containing '!': First, studies show that spammers use second personal pronouns much more than first personal pronouns. In addition, spammers put '!' in their sentences as much as they can to increase impression on users and highlight their reviews among other ones. Reviews are close to each other based on their calculated value, take same values (in [0; 1]).

IV. RESULT AND DISCUSSIONS

Experiments are done by a personal computer with a configuration: Intel (R) Core (TM) i3-2120 CPU @ 3.30GHz, 4GB memory, Windows 7, MySQL 5.1 backend database and Jdk 1.8. The application is web application used tool for design code in Eclipse and execute on Tomcat server. Some functions used in the algorithm are provided by list of jars like opencsv, jsoup and http-components jars etc.

Experimental evaluation results demonstrates the Amazon product review dataset with higher percentage of spam reviews reviews have better performance because when portion of spam reviews builds, probability for a review to be a spam review increments and accordingly result more spam reviews will be classified as spam reviews. The results of the dataset show all the four behavioral features are ranked as first features in the final overall weights. The Fig. 2 graph shows the NetSpam framework features for the dataset have more weights and features for Review-based dataset stand in the second position. Third position belongs to User-based dataset and finally Item-based dataset has the minimum weights (for at least the four features with most weights).

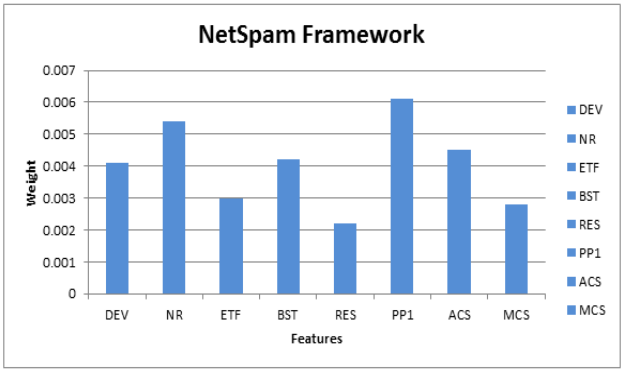

Fig. 2 Feature weights for NetSpam Framework

TABLE I Weights of all features

Features	Weight
DEV	0.0041
NR	0.0054
ETF	0.003
BST	0.0042
RES	0.0022
PP1	0.0061
ACS	0.0045
MCS	0.0028

Fig.3 graph shows the total 510 reviews of amazon single product reviews classified the 185 reviews are spam and 325 reviews are non-spam by using NetSpam framework.

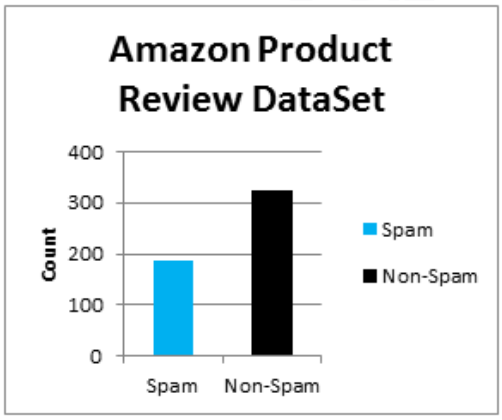

Fig. 3 Spam and Non-spam reviews count

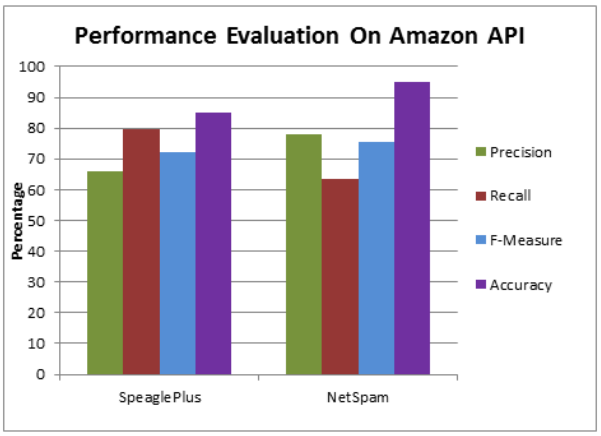

Fig. 4 Performance Analysis between existing and proposed system

The proposed NetSpam framework time complexity is $O(e^2n)$. The netspam framework accuracy is 95.06% which is better than SPaglePlus Algorithm accuracy is 85.14% on using Amazon API for product review dataset.

V. CONCLUSION

This paper proposes a novel spam detection system in particular NetSpam in view of a metapath idea and another graph based strategy to name reviews depending on a rank-based naming methodology. The execution of the proposed structure is assessed by utilizing review datasets. The perceptions demonstrate that ascertained weights by utilizing this metapath idea can be exceptionally powerful in recognizing spam surveys and prompts a superior execution. Furthermore, found that even without a prepare set, NetSpam can figure the significance of each element and it yields better execution in the highlights' expansion procedure, and performs superior to anything past works, with just few highlights. In addition, in the wake of characterizing four fundamental classifications for highlights our perceptions demonstrate that the review behavioral classification performs superior to anything different classifications, regarding AP, AUC and in the

ascertained weights. The outcomes likewise affirm that utilizing diverse supervisions, like the semi-administered strategy, have no detectable impact on deciding the vast majority of the weighted highlights, similarly as in various datasets. Contribution part in this project, applied the Naive Bayes algorithm for sentiment analysis for negative ratio feature's weight calculation. And also for user when searches query he/she will get the top-k product lists as well as one recommendation product item by using personalized recommendation algorithm.

REFERENCES

[1] Ch. Xu and J. Zhang. Combating product review spam campaigns via multiple heterogeneous pairwise features. In SIAM International Conference on Data Mining, 2014.

[2] G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh. Exploiting burstiness in reviews for review spammer detection. In ICWSM, 2013.

[3] A. j. Minnich, N. Chavoshi, A. Mueen, S. Luan, and M. Faloutsos. Trueview: Harnessing the power of multiple review sites. In ACM WWW, 2015.

[4] B. Viswanath, M. Ahmad Bashir, M. Crovella, S. Guah, K. P. Gummadi, B. Krishnamurthy, and A. Mislove. Towards detecting anomalous user behavior in online social networks. In USENIX, 2014.

[5] H. Li, Z. Chen, B. Liu, X. Wei, and J. Shao. Spotting fake reviews via collective PU learning. In ICDM, 2014.