Fake Research Detection using Weighting Algorithm in Netspam Framework

Chaitrali S. Kardile, Revati M. Wahul

Abstract**:- Now a day's our life has become more dependent on social media. Social has opened many opportunity for business so, whenever customer wants to buy new product they will look for other people's opinion. Social media has also have major drawback for business strategies which is spammers. Spammers create spam surveys about various products which mislead a consumer. This online opinion plays important role in business strategies, while positive opinion gives good publicity and market on the other side negative opinion gives bad publicity and market which affects the service providers. To avoid this spammers there have been many research but very have work on user and review related feature. In this investigation we propose a classification system using heterogeneous information network NetSpam framework. This system will classify spam and non-spam reviews using NetSpam algorithm and naïve bayes classifier for sentiment analysis which will provide positive and negative value of the product review. And furthermore if wants to search top product, user can use search query, in addition to that it will display recommendation product on the basis of user's point of interest.

***Index Terms**- Social Media, Amazon API, Spammer, Spam Review, Heterogeneous Information Networks, Naive Bayes, Metapath*

I. INTRODUCTION

Consideration of online reviews or opinions in decision making process has become important. Online reviews plays role of a information resource for purchase decision making, marketing and product designing. Due to profit or fame , imposters have been writing deceptive or fake reviews to promote and\or demote some target product or services. Such imposters are called as review spammers [1]. Positive opinions mean a huge publicity with maximum profit which unfortunately provide strong motive for spammers to post fake reviews. Posting fake reviews results in disrepute business and targeted product. Product which has more positive review attracts more customer's attention than product with less positive reviews. For the marketing purpose and to give more economical benefit to particular product spammers mislead a customer by giving unfair positive reviews of a product and to defame or to damage a reputation of particular business spammer leaves a negative review of a product on social media. Fake review detection has been studied by researcher for multiple times using markov random field [1],positive- unlabeled learning[2], also linguistic patterns, behavioral patterns, graph based algorithm still there is been some aspects are unsolved.

The main objective of NetSpam framework is to use heterogeneous information network(HIN) to build retrieved review data set and convert spam it into classification problem from spam detection problem. In converted HIN review dataset, reach review is connected with each other via different features. To understand the importance of feature weighting algorithm is used, after that this calculated weight used to calculate the very last labels for reviews using both unsupervised and semi-supervised procedures. NetSpam is helpful in identifying importance of feature from the matapath definition and also calculated weight of features. Using feature with more weight leads to indentifying spam with more accuracy and less time complexity.

II. REVIEW OF LITERATURE

In paper [1] author proposes to use Markov Random Field (MRF) to model a reviewers by building a network of appearance of reviewers in burst and apply the Loopy Belief Propagation (LBP) method to identify in case 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. It detects spammers automatically. Disadvantage is: a generic framework is not used for detect spammers.

H. Li has extended a algorithm for classification of group called multi-typed heterogeneous collective classification into collective positive and un-labeled learning [2]. In both PU and non-PU learning environment strong baselines can be increased by F1 scores. Advantages of this model is that this Models only use self-contained features language and can be smoothly generalized from one to another language. It helps to identify fake reviews hiding in the unlabeled reviews that Dianpings algorithm could not indentify.

Author B. Viswanath [3] has used user behavior to classify bad behavior from normal behaviour using unsupervised anomaly detection. 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. It achieves a less than 0.3% false positives with a detection rate more than 66%.

Revised Version Manuscript Received on 16 September, 2019.

Chaitrali S. Kardile, Student, MESCOE, Pune, Maharashtra, India Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu, India.
(Email: *:* kardilechaitrali224@gmail.com)
Revati M. Wahul, Assistant professor, MESCOE, Pune, Maharashtra, India.
(Email: rmwahul@mescoepune.org)

Ch. Xu and J. Zhang [4] proposed to use online product review collection with wise features which can show detailed view of spam campaigns. To cooperate with intuitive and unsupervised wise features author has proposed fraud informer framework. Advantages of this Pair wise features is that to manipulate reviews as per a their best interests it can be ranked in the website globally so that highest rank ones can be found first by using a robust model for finding correlation in colluders. and at the disadvantage it is difficult problem to automate.

M. Crawford elaborates [5] two distinct methods of reducing feature subset size in the review spam domain. The methods include filter-based feature rankers and word frequency based feature selection. After a selecting mostly appeared text in first method it uses chi-squared to rank and select top ranked feature in second method.

A. Djunaidy [6] proposes system ICF++ which uses a text and rating property and it measure the reliability value of a product ,also honesty value of a review along with the trustiness value of the reviewers. Accuracy of this system is better than ICF method. Precision is maximizing.

III. PROPOSED METHODOLOGY

In our system we are using NetSpam framework which is proposed by author Shehnepoor [7]. In this framework, a fresh spam weighting technique is suggested to determine the comparative significance of each feature and show how efficient each feature is in defining spam from ordinary reviews. Alongside NetSpam framework we have used naïve bayes classifier algorithm for sentimental polarity of reviews and Algorithm Top-K-Join-Tuple for the recommendation of product as per user's point of interest.

Proposed framework solves a classification problem by using given dataset as a heterogeneous information network (HIN) [7]. In this review dataset, by using feature and users each node is connected with each other. In NetSpam framework weighting algorithm has been used to calculate weight or a importance of feature, which after that will be used for labeling of review. With the weight of features it calculate the final labeling using supervised and unsupervised methods. To understand how much each feature has contributed in spam detection we have used behavioral and linguistic feature based on user and review.

A. Architecture

Fig.1 demonstrates the architecture of the proposed system. Our suggested system's overall idea is to modify spam detection into classification using dataset as heterogeneous information network. Model review dataset in specific as in which results are linked by distinct kinds of nodes. 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 findings that define two feature opinions.

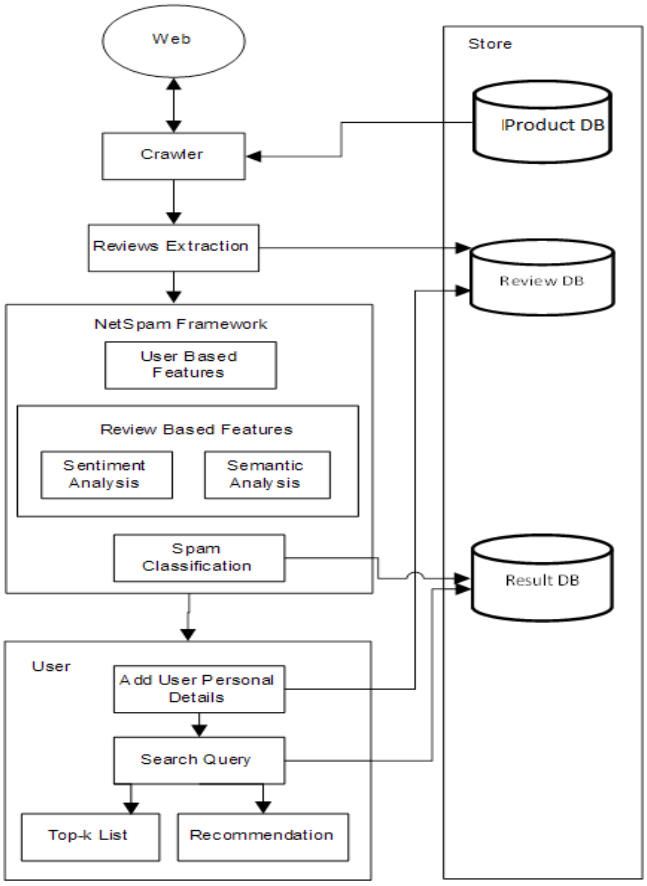

Fig. 1 System Architecture

B. Algorithms

1. Naïve Bayes Classifier Algorithm

Input: Reviews from Amazon API

Output: Sentimental Polarity of reviews

Process:

Step 1: Get the input reviews

Step 2: Assign number of features n[i] where i ranges in 10, 100, 1000, 10000 and 15000.

Step 3: for n in n[i]

Step 3.1: Create wordscore()

Step 3.2: Find_best_words(wordscore,n)

Step 3.3: Evaluate (best_word_features)

Step 4: Create wordscore()

Assign posword[j] and negword[k] Split the words by removing punctuations

Build frequency distribution of all words

Step 5: Find_best_words(wordscore,n)

Find number of positive, negative and total number of words

Build dictionary of the wordscore based on the Chi-square test (i.e.) word_score[t]

By sorting the wordscore, bestwords are found

Step 6: Evaluate (best_word_features)

Assign posfeatures[j] and negfeatures[k]

Split the sentences into individual words.

Select ¾ of the features for training and ¼ of the features for testing

Train using Naïve Bayes Classifier

Step 7: Reviews are classified as positive and negative based on the score words

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: $m_{u,v}^{Pl}$ :  value of metapath
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. Features

We have used the notion of metapath to create the following connection between reviews. A metapath is defined as a route between two reviews, indicating the link between two reviews by sharing characteristics. Refer to its overall definition, which is information on information, when talking about metadata. In our case, the data is the review in text, and metadata means the collection of review data, including the user as a individual who wrote a review, the business or service provider for whom the review has been written, the rating given to a product, a date on which review is uploaded, and label to a review of being spam or genuine.

Metapath is created using following features:-

i. User Behavioral

This features is about the each individual who is posting a review as a user. We have used this feature to categorize all the reviews which are written by particular individual. This feature has to divided in two categorize that are burstiness and negative ratio. Burstiness is used to identify review written by single user in short period of time. Negative ratio is destructive reviews or ratin with low scores which are posted to defame competitive business.

ii. User Linguistic

This feature is derived from the users feelings or opinion about particular product or service provider. This feature is categorized in Average Content Similarity (ACS) and Maximum Content Similarity (MCS). Spammers generally write reviews with same template to avoid time wasting and as a result they have same reviews. This feature requires semantic analysis to be performed to detect copy paste mechanisms used by spammers. The copy paste reviews written by spammers can be identify by calculating time between their start and end of the posted review because to post fake review with many word take less time than original posted review.

iii. Review Behavioral

Metadata of the review is used in this feature to identify spammers. This feature is categorized review on basis of early time frame and rate deviation. In early time frame fake reviewers or spammers try to write their review in short period of time to keep it in recent reviews or in top reviews. Rate deviation is used to identify spammers on basis of rating or high scores. To get more publicity or business spammers rates high a particular business. As a result, businesses gets variation in scores also high variance and deviation which affects the economy of business.

iv. Review Linguistic

Extracted text of a reviews used in this feature. Review linguistic is categorized into First Personal Pronouns(PP1) ratio and also Ratio of exclamation mark. When the spammer uses '!' in sentence and second pronoun to attract more users attention and make impression.

IV. RESULT AND DISCUSSIONS

To implement this system we have used personal computer with basic hardware requirement and mysql 5.1 backend database and JDK 1.8. This web based application has used Eclipse Luna and Tomcat Server to design code.

Experimental evaluation results show that the Amazon product review dataset has better performance with a maximum percentage of spam reviews because when small segment of spam reviews builds, the likelihood of a review being a spam review increment will result in more spam reviews being classified as spam reviews. The Fig. 2 graph shows the NetSpam framework features in which first position, the dataset have more weights and features based on Review dataset stand in the second position. User and item based dataset stands in third and fourth position respectively with the minimum weights. 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.

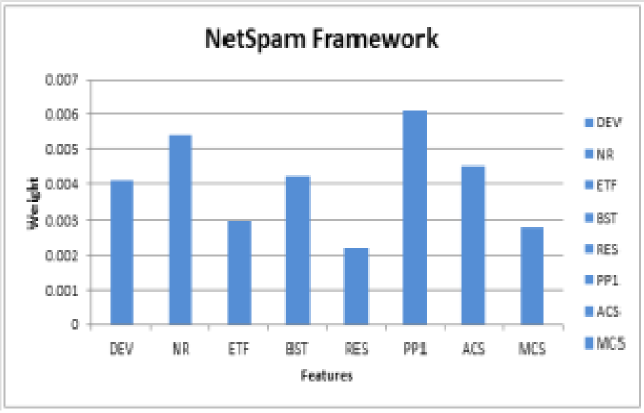

Fig. 2 Feature weights for NetSpam Framework

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

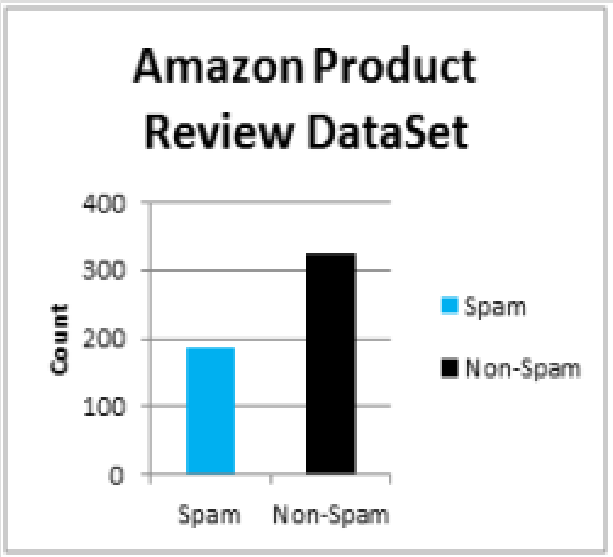

Fig. 3 Spam and Non-spam reviews count

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.

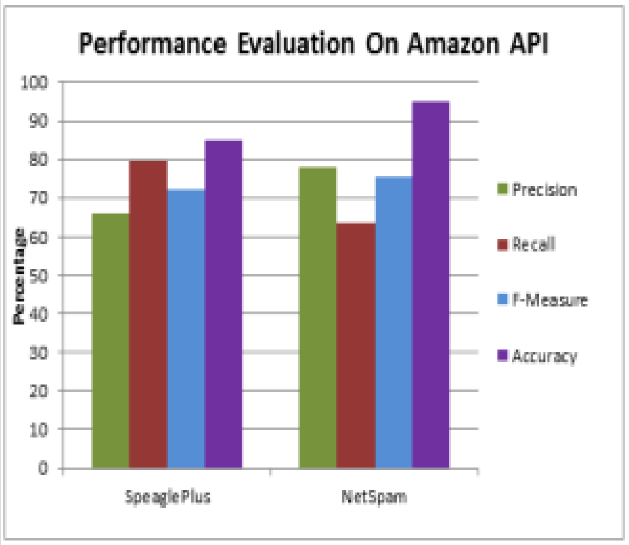

Fig. 4 Performance Analysis between existing and proposed system

V. CONCLUSION

This paper proposes a 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. 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.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

REFERENCES

1. G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh. Exploiting burstiness in reviews for review spammer detection. In ICWSM, 2013.
2. H. Li, Z. Chen, B. Liu, X. Wei, and J. Shao. Spotting fake reviews via collective PU learning. In ICDM, 2014.
3. 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.
4. Ch. Xu and J. Zhang. Combating product review spam campaigns via multiple heterogeneous pairwise features. In SIAM International Conference on Data Mining, 2014.
5. M. Crawford, T. M. Khoshgoftaar, and J. D. Prusa. Reducing Feature set Explosion to Faciliate Real-World Review Sapm Detection. In Proceeding of 29th International Florida Artificial Intelligence Research Society Conference. 2016.
6. E. D. Wahyuni and A. Djunaidy. Fake Review Detection From a Product Review Using Modified Method of Iterative Computation Framework. In Proceeding MATEC Web of Conferences. 2016.
7. Shehnepoor, Saeedreza & Salehi, Mostafa & Farahbakhsh, Reza & Crespi, Noel. (2017). NetSpam: A Network-Based Spam Detection Framework for Reviews in Online Social Media. IEEE Transactions on Information Forensics and Security. PP. 1-1. 10.1109/TIFS.2017.2675361.