NetSpam: A Network-Based Spam Detection Framework for Reviews in Online Social Media

Ms. Sunanda[1], Mrs.Subiya Noorian[2], Dr. Shivanand S. Rumma[3], Sweta[4]

[1,2]PG Scholar, Dept.of P.G.Studies and Research in Computer Science, Gulbarga University, Kalaburagi.
[3]Chairman, Dept.of P.G.Studies and Research in Computer Science, Gulbarga University, Kalaburagi.
[4]Faculty, Dept.of P.G.Studies and Research in Computer Science, Gulbarga University, Kalaburagi.

Abstract

"NetSpam: A Network-Based Spam Detection Framework for Reviews in Online Social Media" refers to unwanted or unsolicited messages, posts, or emails sent over the internet or a network. It's essentially digital junk mail.

This system ensures about irrelevant or promotional comments on blogs, forums, or social media. In today's digital age, online reviews play a crucial role in shaping consumer decisions and business reputations. However, the open nature of these platforms also provides opportunities for malicious actors to post spam reviews, misleading users and undermining the credibility of genuine feedback. Social media enables customers to share their views, opinions and experiences as product reviews. These product reviews facilitate customers in buying quality products. Due to the significance of online reviews, fake reviews, commonly known as spam reviews are generated to mislead the potential customers in decision-making. To cater this issue, review spam detection has become an active research area.

Keywords: Heterogeneous Information Network (HIN), Online Social Media, Social network, Spammers, Spam detection, Spam reviews.

The existing system shows that spamblog outperforms the existing methods and among four categories of features; including review-behavioural, user behavioural, review linguistic, user linguistic, the first type of features performs better than the other categories. The general concept of this proposed framework is to model a given review dataset as a Heterogeneous Information Network (HIN) and to map the problem of spam detection into a HIN classification problem. This utilizes spam features for modelling review datasets as heterogeneous information networks to map spam detection procedure into a classification problem in such networks. Measure of the computational time or effort required by an algorithm to complete is more.

The primary challenge lies in accurately identifying and filtering out these spam reviews from genuine ones. Traditional spam detection methods often fall short due to the evolving tactics of spammers and the complexity of distinguishing between authentic and fake reviews. This inadequacy can lead to a loss of trust in online review platforms and potential economic losses for businesses.

1. PROPOSED SYSTEM

In the proposed system spamblog is able to find features importance even without ground truth, and only by relying on metapath definition and based on values calculated for each review. Spamblog improves the accuracy compared to the state of the art in terms of time complexity, which highly depends to the number of features used to identify a spam review; hence, using features with more weights will resulted in detecting fake reviews easier with less time complexity.

A new Content Based Algorithm 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.

Advantage of Proposed System

To identify spam and spammers as well as different type of analysis on this topic. Written reviews also help service providers to enhance the quality of their products and services.

2. Methodology Used

The general methodology in developing a system is involved in different phases, which describe the system's life cycle model for developing software project.

Those are,

1. Requirement analysis phase
2. Design phase
3. Development phase
4. Coding phase

System design is the process of designing the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system.

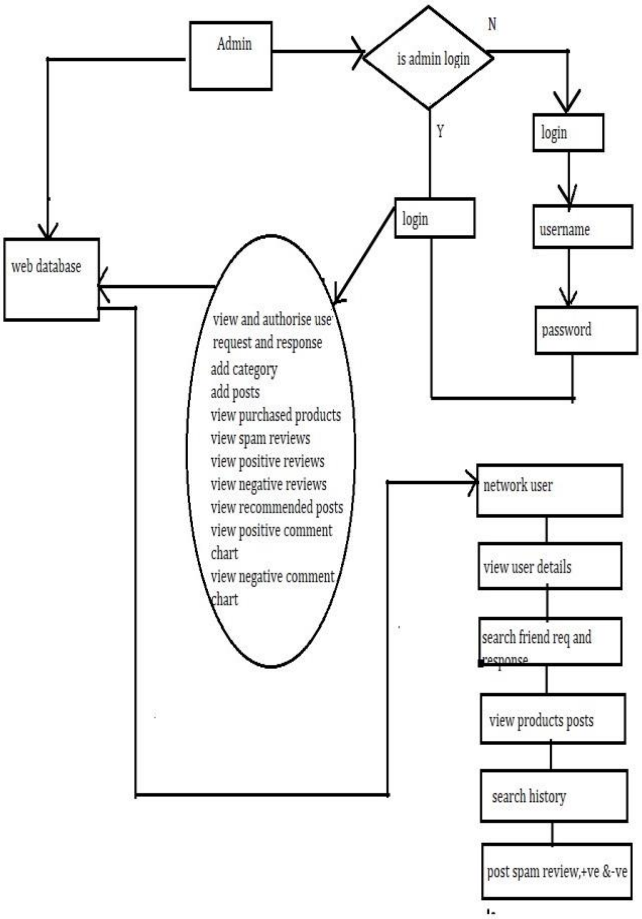


Fig 1:flow diagram of proposed system

3. Module Description

1. Admin

In this module, the admin has to login by using valid user name and password. After login successful he can do some operations, such as adding Categories, Adding Products for that Categories, Viewing and authorizing users, View Spam accounts details, viewing friend request and response, all recommended posts, all posts with all Reviews, All Positive and Negative Reviews, Removing Products, Viewing All Purchased Products, viewing Positive and Negative Reviews Chart on products.

2. User

In this tab, there are ‘n’ numbers of users are present. user should register before doing any operations. once user registers, their details will be stored to the database. after registration successful, he has to login by using authorized user name and password. once login is successful user will do some operations like viewing their profile account details like spam or normal, search users and send friend request, viewing friend requests, searching posts and recommend to friends and viewing all product recommendations sent to him by his friends, commenting on posts, purchasing products and viewing their product search history.

Hypertext Markup Language (HTML):

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows
Users to produces web pages that include text, graphics and pointer to other web pages (Hyperlinks).

Java Database Connectivity

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

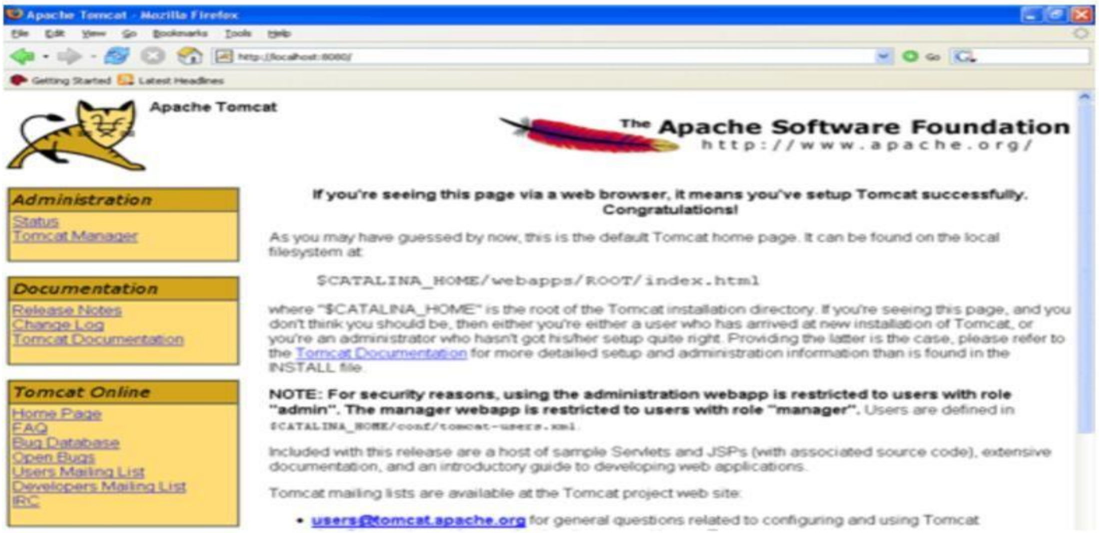
Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere.

Simply put, JDBC makes it possible to do three things:

- Establish connection with database
- Send SQL Statements
- Process the result

Tomcat 6.0 web server

Tomcat is an open-source web server developed by apache group. apache tomcat is the servlet container that is used in the official reference implementation for the java servlet and java server pages technologies. The java servlet and java server pages specifications are developed by sun under the java community process. Web servers like apache tomcat support only web components while an application server supports web components as well as business components (BEAs Web logic, is one of the popular application servers). To develop a web application with jsp/servlet install any web server like JRun, Tomcat etc to run your application.



If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:

```
$CATALINA_HOME/webapps/ROOT/index.html
```

where "$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the Tomcat Documentation for more detailed setup and administration information than is found in the INSTALL file.

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in `$CATALINA_HOME/conf/tomcat-users.xml`.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation, and an introductory guide to developing web applications.

Tomcat mailing lists are available at the Tomcat project web site:

- users@tomcat.apache.org for general questions related to configuring and using Tomcat

Fig 2:Apache Tomcat Tool

MYSQL:

MySQL is an open-source relational database management system. The abbreviation for SQL is Structured Query Language. SQL is a language programmer use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computers storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is written in C and C++. Its SQL parser is written in Yac, but it uses a home-brewed lexical analyser. MySQL is offered under two different editions: the open-source MySQL community server and the proprietary enterprise server. MySQL enterprise server is differentiated by a series numbering system and is built from the same code base.

MySQL can be built and installed manually from source code, but it is more commonly installed from a binary package unless special customizations are required. On most Linux distributions, the package management system can download and install.

5. TESTING APPROACHES

1. Bottom-up approach
2. Top-down approach

Bottom-up Approach:

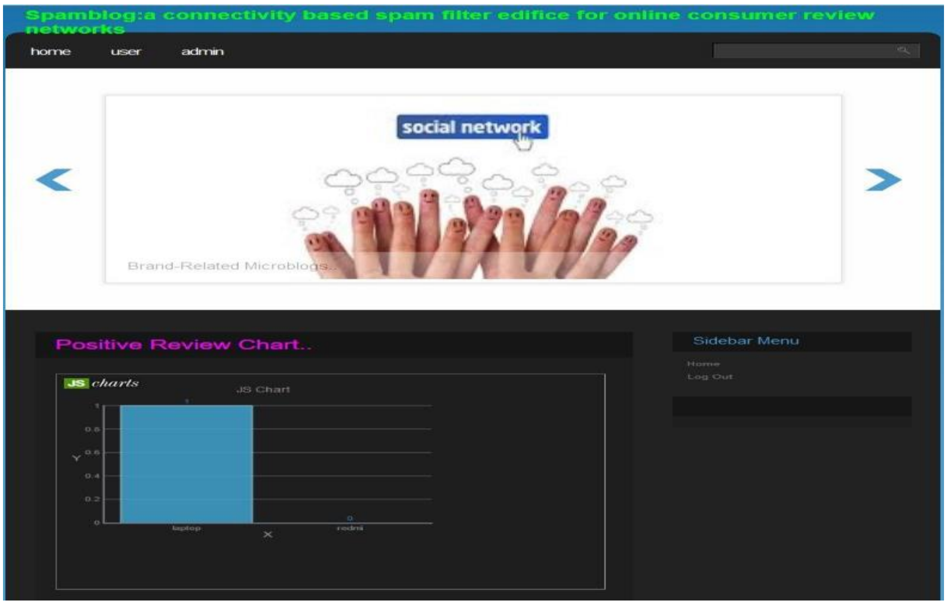
Testing can be performed starting from smallest and lowest level modules and proceeding one at a time. For each module in bottom up testing a short program executes the module and provides the needed data so that the module is asked to perform the way it will when embedded within the larger system. When bottom level modules are tested attention turns to those on the next level that use the lower-level ones they are tested individually and then linked with the previously examined lower level modules.

Top-down approach:

This type of testing starts from upper-level modules. Since the detailed activities usually performed in the lower-level routines are not provided stubs are written. A stub is a module shell called by upperlevel module and that when reached properly will return a message to the calling module indicating that proper interaction occurred. No attempt is made to verify the correctness of the lower-level module.

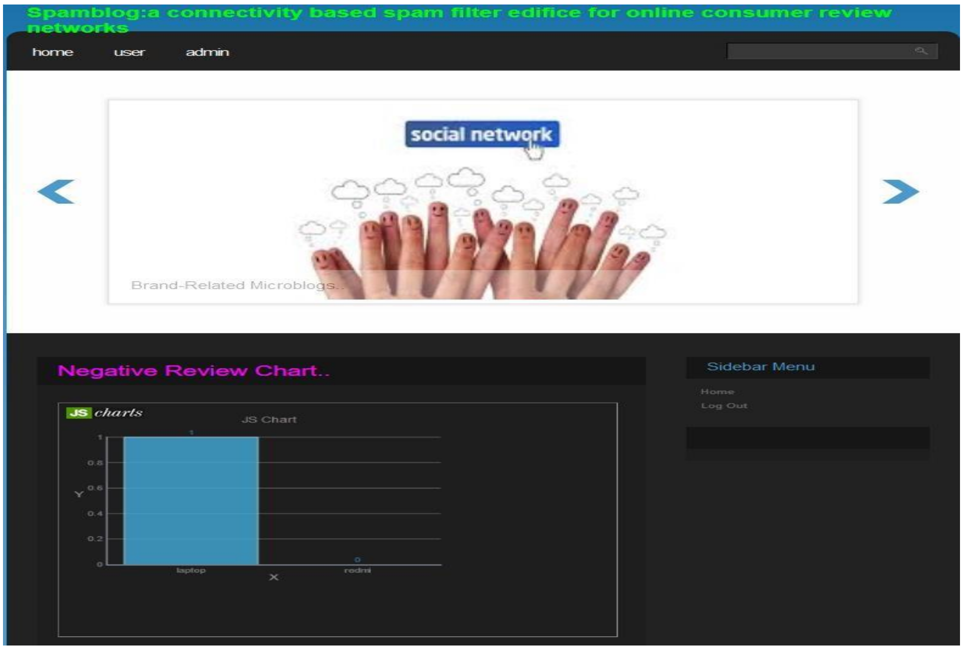
6. Results/Report

The proposed system aims to detect and mitigate spam reviews, and its results can be evaluated through positive review chart and negative review chart.



Spamblog:a connectivity based spam filter edifice for online consumer review networks

Screenshot showing Positive Comment Chart



Spamblog:a connectivity based spam filter edifice for online consumer review networks

Screenshot showing Negative Comment Chart

7. Conclusion

This proposed system introduces a spam detection framework namely spamblog. The performance of the proposed framework is evaluated by using two real-world labelled datasets of Yelp and Amazon websites. It shows that calculated weights by using this Metapath concept can be very effective in identifying spam reviews and leads to a better performance. In addition, the system found that even without a train set, spamblog can calculate the importance of each feature and it yields better performance in the features' addition process, and performs better than previous works, with only a small number of features. Moreover, after defining four main categories for features this system show that the reviews behavioural category performs better than other categories, in terms of SAP, AUC as well as in the calculated weights. The results also confirm that using different supervisions, similar to the semi-supervised method, have no noticeable effect on determining most of the weighted features, just as in different datasets.

8. References

1. J. Donfro, A whopping 20 % of yelp reviews are fake. Accessed: 2015-07-30.
2. M. Ott, C. Cardie, and J. T. Hancock. Estimating the prevalence of deception in online review communities. In ACM WWW, 2012.
3. M. Ott, Y. Choi, C. Cardie, and J. T. Hancock. Finding deceptive opinion spam by any stretch of the imagination. In ACL, 2011.
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. N. Jindal and B. Liu. Opinion spam and analysis. In WSDM, 2008.
6. F. Li, M. Huang, Y. Yang, and X. Zhu. Learning to identify review Spam.Proceedings of the 22nd International Joint Conference on Artificial Intelligence; IJCAI, 2011.
7. G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh. Exploiting burstiness in reviews for review spammer detection. In ICWSM, 2013.
8. j. Minnich, N. Chavoshi, A. Mueen, S. Luan, and M. Faloutsos. Trueview: Harnessing the power of multiple review sites. In ACM WWW, 2015.
9. Viswanath, M. Ahmad Bashir, M. Crovella, S. Guah, K. P. Gummadi, B. Krishnamurthy, and A. Mislove. Towards detecting anomalous user behaviour in online social networks. In USENIX, 2014.
10. L. Akoglu, R. Chandy, and C. Faloutsos. Opinion fraud detection in online reviews bynetwork effects. In ICWSM, 2013.
11. R. Shebuti and L. Akoglu. Collective opinion spam detection: bridging review networks and metadata. In ACM KDD, 2015.
12. S. Feng, R. Banerjee and Y. Choi. Syntactic stylometry for deception detection. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers; ACL, 2012.
13. N. Jindal, B. Liu, and E.-P. Lim. Finding unusual review patterns using unexpected rules. In ACM CIKM, 2012.
14. E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw. Detecting product review spammers using rating behaviours. In ACM CIKM, 2010.
15. A. Mukherjee, A. Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos, and R. Ghosh. Spotting opinion spammers using behavioural footprints. In ACM KDD, 2013.
16. S. Xie, G. Wang, S. Lin, and P. S. Yu. Review spam detection via temporal pattern discovery. In ACM KDD, 2012.
17. G. Wang, S. Xie, B. Liu, and P. S. Yu. Review graph based online store review spammer detection. IEEE ICDM, 2011.