A Personalized Method for Blacklisting Users and Implementing Filtered Walls in OSN

Sai Sudha Gadde

M.Tech. Student,Andhra Loyola Institute of Engineering and Technology, Vijayawada.

Abstract-OSN (Online Social Networks) may also come under one of the platforms where the wide range of communication and sharing of information is taking place since last few years. But still there must be some privacy regarding the preferences of the user. As social networks are very flexible in sharing or posting information, there has to be some limitation regarding the data being posted on the user's wall. The choice of separating the unwanted data from the user is left to the opinion of the user, in this case.

The concept of Content-Based Filtering and Blacklisting are the main focus in this paper. An Online Setup Assistant for Filtering rules Threshold is also maintained in order to get the number of times this filtered word is hit by the users.

Mainly, the user is able to even add the information to the filter which user wants to avoid in the messages posted on his/her wall. Even if required future corrections are also available in this case, like updating of the filter or even is privileged to make a correction in future by giving user an option of removing the filter in case if the user feels it is necessary to.

Keywords: Social Network Applications, Content-Based Filtering, Filtering Rules, Blacklisting, Online setup Assistant for Filtering rules Threshold.

I. INTRODUCTION

As a continuation to the paper published in DEC-2014, Issue 12, "A Novel method for Providing Discrete Filtering and Blacklisting Options for OSN Users" published in IJSEAT.

In the previous paper the elaborated discussion of the main basic topics and their implementation is mentioned.

Now in this paper, the further developments and additions to improve the performance of the project are to be discussed. These developments as a brief include addition of new modules, improved and flexible logic in filtering techniques and also representing the dynamic graph forms of the dynamically changing data. The logic used in this paper is developed by assessing the need of the user and the use of the logic implemented.

1. The Filtering Rules which are formulated according to the need of the application are improved by considering its applicability option.
2. The Filtered word not only the word alone, even if it contains any other punctuation marks or symbols like !,@,#,$,%,^,&,*,(,),_,-,+,=,.,',,'s,s …etc at the end, then also the word will be blocked and considered as occurrence of filtered word.
3. Blacklist option accessed as an attribute in an entity, but not made as a separate entity.
4. Admin module is added in this implementation to just view some of the data of the users, graph/number of the filters added by users, graph/number of the

occurrences by the filtered word encountered by means of the five categories (Vio - Violence, Vul - Vulgar, Off-Offensive, Hat-Hate, Sex), these are the major categories under which the filter added by the user can be categorized.

5. Dynamic Graph and the values table for the added filters under these above mentioned categories.
6. Dynamic Graph and the values table for the occurrences of the filtered words while posting the comments. This work is also based on the five categories of words.

These are the changes that are implemented in this paper, that are to be explained in detail in further sections.

The remainder of this paper is organized as follows:
Section 2 Overview of the system,
where as Section 3 introduces the concept of the proposed system and
Section 4 reviews the implementation process. Finally, Section 5 concludes the paper.

II. SYSTEM OVERVIEW

In the proposed system of M.Vanetti[3], the problem can be solved by giving the user direct control over the messages posted on their wall, with the help of flexible rule based system, and a Machine Learning based Soft Classifier.

But in actual case as mentioned in some other papers it is more difficult to find out what a user wants and what a user doesn't want to get. This may also change by time as there may happen some considerable changes regarding the user's likes and dislikes. So as of now there is no system that can itself know what a user wants, without user interaction.

In the case where user interaction is needed also may be sometimes a hectic task for the user to take up on. Some ways where such interactive knowledge, made by question airing the user can be used for avoiding the unwanted data to some extent. But in this case also there are some loopholes in the cases where user cannot afford that much time in answering the questions posted by the system. Also in some cases, users may lack in patience to answer to all the questions posed by the system.[7]
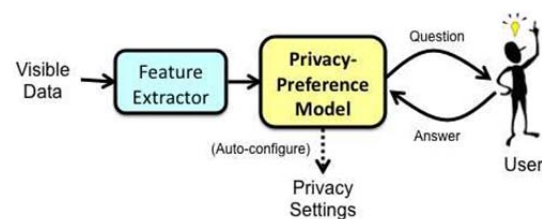

Figure 1: Privacy Preference Model

As mentioned in the paper [10], even short text classification also may not be an easy and effective way of avoiding unwanted messages. In that case, a bag of words need to be maintained regardless of the user's opinion and those words are to be checked each time when a user writes a comment and tries to post it.

Also a nearer approach to this paper is proposed in paper [6] and [3]. Both are almost similar to the concept chosen in this paper.
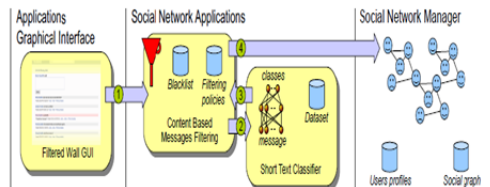

Figure 2: Filtered Wall Conceptual   Architecture

With the extension and small changes to the main concept, this paper comes out with the solution for the problem of avoiding unwanted messages on OSN walls. This paper gives the overall idea by which it is implemented and how this filtering and concept of blacklisting is made possible.

There are 3 modules in this system, they are:
   a) User Module.
   b) Online Setup for Filtering rules Threshold.
   c) Admin Module.

All these modules are considered equally important, as one module may help the other module in linking its path for the right use of the system implementation process in a useful manner.

Figure 3: Overview of the System

III. PROPOSED SYSTEM

In this section the actual discussion about how a filtering rule is formulated has been discussed in the earlier paper.

Now in this paper the improvised techniques adopted for more usability are to be discussed in this section.

1. In brief, according to this system Filtering rule is a tuple of (CreatorSpec, Word, Type, Applicability)
   a. **CreatorSpec** is the specification of the user who creates this filter.
   b. **Word** indicates the word which should be filtered in the message and by default if the word is found

it gives a warning message to the user trying to post it.
   c. **Type** is the category under which the word comes under.
   d. **Applicability** means the filter to be applied to a single user or ALL the friends of the user.

Here the option given as "Applicability" mentioned above plays a multirole in applying the filter in specific or as a whole.

- **ALL:** If this option is selected, the filter is applied to all the friends of the user who adds the filter. Even if any new friends are added later, this will be applied to them also.
- **Friends-List:** Other than ALL, all the listed friends of the user are given, and any one can be selected. The selected friend will be applied for the particular filter.

2. The word added to the filter may be any word disliked by the user to be posted on their wall, but the other forms like any special characters and symbols succeeding the word is also blocked and not allowed.

For example, let the word to be filtered be **kill**, so that any message long or short which contains the word **kill** in the content will be blocked from being posted on the destination user's wall. Let us observe some different cases of messages where whether they are accepted by this filtering rule or not.

Case 1: May I know which remedy can i use to <u>kill</u> mosquitoes?   - blocked

Case 2: You are very s<u>kill</u>ful   - posted

Case 3: Mom <u>KILLS</u> mosquitoes using coils - blocked

Similarly, if the message contains other characters succeeding the word like 'kill.', 'kill!', 'KILL*'... etc. The filter word is also case sensitive, even it may be given in any case upper, lower or mix of both, the filter is able to detect the word and filter it.

3. The other improvisation can be labelled as Blacklist option. In this system, Black list is not considered as a separate entity to provide logic to the system implementation, rather it is just an attribute that is appended to a table and can give more detail information regarding the blacklisted and blacklisting user.

Friends are an entity with attributes as (sender, sender_id, getter, getter_id, blist, date). Each time when a user1 tries to add user2 as friend then a record is inserted as user1 as sender and user2 as getter, by default blist=no is entered.

In the similar way when the user2 adds user1 as friend then another record mentioning the user2 as sender and user1 as receiver and blist=no is added to friends entity.

Here blist is a variable whose value is by default 'no', and when once the user1 adds user2 to blacklist then blist=yes is made in the record where user1 is sender and user2 is getter.

In this way, by the help of one attribute itself the working of blacklist is made possible.

4. In this system Admin module is introduced, to just monitor how the users of the network are making use of the filtering system and also to get dynamic graphs regarding

the added filters and also the word occurrences of the filtered words.

Admin is given information regarding the users, number of filters added category wise and its graph representation, number of occurrences of the word filtered in messages tried to post and its graph representation.

5. A Dynamic graph related to the number of words added as filters, also arranged category wise.

Each category is shown individually in the graph (pie-chart) and also the individual count is also given separately for all the categories in a table format. Each column representing the category and all its count of the words which are added under the particular category.

6. This can also be considered as the online setup for filtering rules threshold. The graph related to the number of filtered words occurred while user tries to post a comment, also this result is arranged category wise.

Each category is shown individually in the graph (pie-chart) and also the individual count is also given separately for all the categories in a table format. Each column representing the category and all its count of the filtered words occurrences category wise which are added under the particular category.

These are the changes and improvements done regarding the system compared to the previous one. In the next section the actual picture of the system can be obtained

IV. IMPLEMENTATION PROCESS

In this system we have 3 modules. Namely,
1. User Module.
2. Filtering rules Threshold Values.
3. Admin Module.

User Module

In User Module, the following functionalities are provided to the user regarding the issue of filtering and blacklisting:

   i. Adding word to Filter.
   ii. Chance of Updating Existing Filter.
   iii. Chance of Removing Existing Filter.
   iv. Adding Existing friend to Blacklist.
   v. Removing Blacklisted friend from Blacklist.

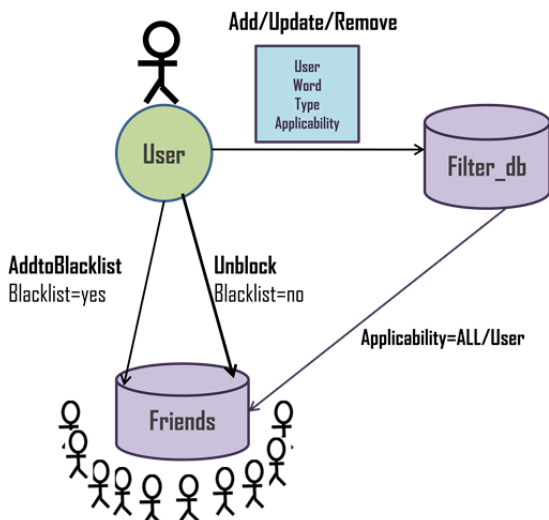All these functions are shown in the figure below :

Figure 4: Options for User in User Module

Filtering rule Threshold Values

This module is the representation of the resulting final totals of the number of times the word added in the filter occurred and as a whole, all the categories are individually totalled and represented in the form of table and a pie-chart which is generated dynamically.

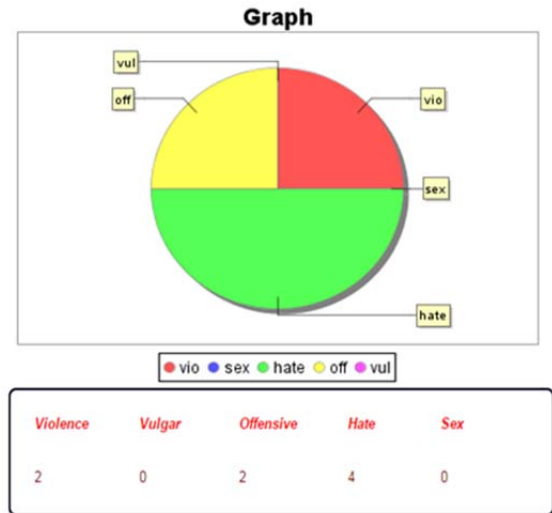| Violence	Vulgar	Offensive	Hate	Sex
2	0	2	4	0

Figure 5: Representation of Word Occurrences

Also there is a dynamically generated pie-chart generated for the values of the total added filters as a whole categorized individually, and showed the values in the format of the table and also pie-chart

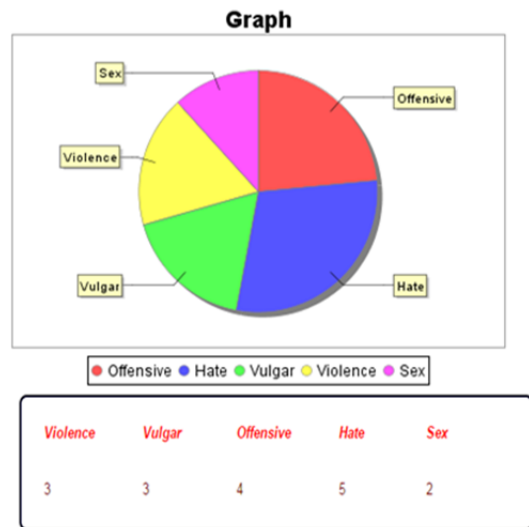| Violence	Vulgar	Offensive	Hate	Sex
3	3	4	5	2

Figure 6: Representation of Added filters

Admin Module

In the Admin Module, the admin has permission to view some extent of the user data, graphs generated in the system.

   i. All Users Data to some extent.
   ii. Graph and table for added filters.
   iii. Graph and table for filtered word occurrence.

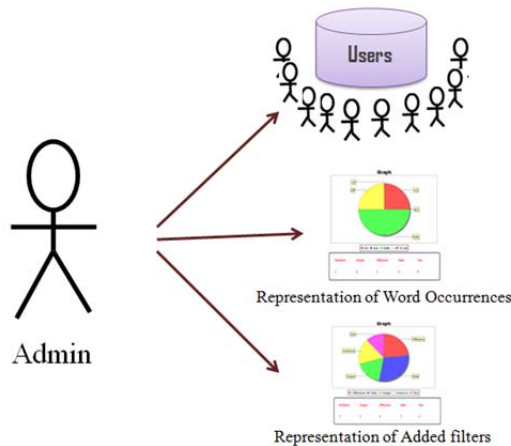The figure below can clearly show what are the content visible to the admin.

Figure 7: Visible data to Admin Module

So the Admin Module is just inorder to retrieve different graphs generated and also to have a glance at all users of the network.

This is about different modules of the system implemented and showed in detail in this paper.

V. CONCLUSION

In this paper, just the improved version is discussed for the better results and usability for the system. As the purpose of Filtering messages and Blacklisting is satisfied, this system is suitable for the current problem.

But there can be chances of developing a better system with much better technologies evolving in the near future, depending upon the need of future generation systems, also can enhance the performance of the system in terms of speed and other regards.

So, finally i would like to conclude that the current system is suitably developed according to the present needs and generation. All the required functionalities are also implemented and are likely suitable to the present scenario.

REFERENCES

[1] P.W. Foltz and S.T. Dumais, "Personalized Information Delivery: An Analysis of Information Filtering Methods," Comm. ACM, vol. 35, no. 12, pp. 51-60, 1992.
[2] N.J. Belkin and W.B. Croft, "Information Filtering and Information Retrieval: Two Sides of the Same Coin?" Comm. ACM, vol. 35, no. 12, pp. 29-38, 1992.
[3] Marco Vanetti, Elisabetta Binaghi, Elena Ferrari, Barbara Carminati, and Moreno Carullo," A System to Filter Unwanted Messages from OSN User Walls" IEEE, VOL. 25, NO. 2, FEBRUARY 2013.
[4] R.J. Mooney and L. Roy, "Content-Based Book Recommending Using Learning for Text Categorization," Proc. Fifth ACM Conf. Digital Libraries, pp. 195-204, 2000.
[5] F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, vol. 34, no. 1, pp. 1-47, 2002.
[6] M. Vanetti, E. Binaghi, B. Carminati, M. Carullo, and E. Ferrari, "Content-Based Filtering in On-Line Social Networks," Proc. ECML/PKDD Workshop Privacy and Security Issues in Data Mining and Machine Learning (PSDML '10), 2010.
[7] L. Fang and K. LeFevre, "Privacy Wizards for Social Networking Sites," Proc. 19th Int'l Conf. World Wide Web (WWW '10), pp. 351- 360, 2010.
[8] M. Chau and H. Chen, "A Machine Learning Approach to Web Page Filtering Using Content and Structure Analysis," Decision Support Systems, vol. 44, no. 2, pp. 482-494, 2008.
[9] Hanani, U., Shapira, B., Shoval, P.: Information filtering: Overview of issues, research and systems. User Modeling and User-Adapted Interaction 11, 203–259 (2001).
[10] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M.Demirbas, "Short Text Classification in Twitter to Improve Information Filtering," Proc. 33rd Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '10), pp. 841-842, 2010.
[11] Manning, C., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge, UK (2008).
[12] P.J. Denning, "Electronic Junk," Comm. ACM, vol. 25,no. 3, pp. 163-165, 1982.

Gadde. Sai Sudha, is a student of Andhra Loyola Institute Of Engineering And Technology, Vijayawada. Under JNTUKakinada. She received her B.Tech in Information Technology from Andhra Loyola Institute Of Engineering and Technology, Vijayawada .Presently she is pursuing her M.Tech (Computer Science & Engineering) from this college. Her area of interest includes Data Mining, Data Base Management Systems and Object oriented Programming languages, some current trends and techniques in Computer Science.Published a paper in Data Mining .

.