INTERNATIONAL EDUCATIONAL APPLIED RESEARCH JOURNAL (IEARJ)
Volume 03, Issue 09, Sep 2019
E-ISSN: 2456-6713

A HYBRID APPROACH FOR DETECTING AUTOMATED SPAMMERS IN TWITTER

***Ramu Kancherla,[1] D. Murali[2]**

[1]*M.tech Student, Computer Science and Engineering, Brahmaiah College of Engineering, Nellore, AP, India, 524366.*

[2]*Associate Professor, Computer Science and Engineering, Brahmaiah College of Engineering, Nellore, AP, India, 524366.*

Abstract: *Twitter is one of the most popular micro blogging services, which is generally used to share news and updates through short messages restricted to 280 characters. However, its open nature and large user base are frequently exploited by automated spammers, content polluters, and other ill-intended users to commit various cybercrimes, such as cyber bullying, trolling, rumor dissemination, and stalking.*

The novelty of the proposed approach lies in the characterization of users based on their interactions with their followers given that a user can evade features that are related to his/her own activities, but evading those based on the followers is difficult. Nineteen different features, including six newly defined features and two redefined features, are identified for learning three classifiers, namely, random forest, decision tree, and Bayesian network, on a real dataset that comprises benign users and spammers.

Keywords: *website, new designs, high, security, new technology.*

1. INTRODUCTION:

TWITTER, a micro blogging service, is considered a popular online social network (OSN) with a large user base and is attracting users from different walks of life and age groups. OSNs enable users to keep in touch with friends, relatives, family members, and people with similar interests, profession, and objectives. In addition, they allow users to interact with one another and form communities. A user can become a member of an OSN by registering and providing details, such as name, birthday, gender, and other contact information. Although a large number of OSNs exist on the web, Facebook and Twitter are among the most popular OSNs and are included in the list of the top 10 websites1 around the worldwide.

1.1 OSN and the Social Spam Problem:

The main contributions of this study can be summarized as follows:

- A novel study that uses community-based features with other feature categories, including *metadata*, *content*, and *interaction*, for detecting automated spammers.

- A detailed analysis of the working behavior of automated spammers and benign users with respect to newly defined features. In addition, two-tailed *Z*-test statistical significance analysis is performed to answer the following question: "*is the difference between the working behavior of spammers and benign users in terms of newly defined features a random chance?*"

- A thorough analysis of the discriminating power of each feature category in segregating automated spammers from benign users.

2. LITERACY SURVEY:

2.1 Introduction:

It has becomes quite simple to access information from all around the world with the help of Internet. Increase in popularity of social networking sites allows us to gather enormous amount of data and information about users, their relationships, friends and family. Large amount of data present on these sites attracts also malicious users. Such users use autonomous programs that act like human to steal the user's personal information, spreading misinformation and propaganda. These special programs are called social bots.

Spam detection is critical task for security of social media. It is very important to identify spam in online social network in order to protect users from various kinds of attacks. In social networking sites, information is shared on trust relationships. Usually it is shared among personal friends and it might be public as well, i.e., it can be accessed by everyone. Some users have tendency to accept unknown friend request to become popular at the cost of his privacy. Network trust is very important as far as security perspectives are concerned.

2.2 Online Social Network:

Online Social Network Vulnerabilities Large number of users and huge amount of information being shared increases security and privacy issues in online social networking sites (OSNs). According to statistics released by Facebook 655 million user log on to this site and share 4.75 billion pieces of information with each other . Large amount of data present on these sites attract the malicious groups. These groups use autonomous programs that act like human to seal the user's personal information, spread misinformation and propaganda. These spacial programs are called social bots. For example, someone using social engineering to hack computer network might try to gain the confidence of an official user and get them to disclose information that compromises the network's security. Social engineers often rely on the natural helpfulness of people as well as on their weaknesses. They may call the authorised employee with some kind of urgent problem that requires immediate network access.

2.3 The Social Engineering Attack Cycle:

Clone attacks are also bigger threats in which attacker creates clone profiles of user's friends, if person accept the request then all the information will be accessible to the attacker. Twitter has two million users in one month which share 8.3 million tweets per hour. Twitter limit user to post their messages (tweets) up to 140 characters.

3. SYSTEM ANALYSIS

3.1 Existing System:

Sahami et al. proposed textual and non-textual and domain-specific features and learned naive Bayes classifier to segregate spam emails from legitimate ones. Schafer proposed metadata-based approaches to detect botnets based on compromised email accounts to diffuse mail spams. Spam campaigns on Facebook were analysed by Gao et al. using a similarity graph based on semantic similarity between posts and URLs that point to the same destination.

3.2 Disadvantages:

- There are no Hybrid techniques to classify different spam's behaviours.
- There is no spam bot detection techniques.

3.4 Advantages:

A novel study that uses community-based features with other feature categories, including metadata, content, and interaction, for detecting automated spammers.

Three key considerations involved in the feasibility analysis are:

- Economical Feasibility
- Technical Feasibility
- Social Feasibility

3.5.1 Economical Feasibility:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited.

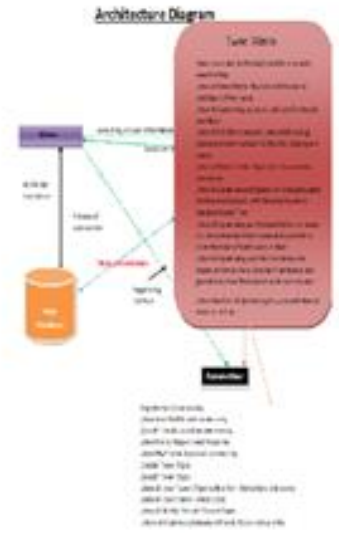
4. SYSTEM REQUIREMENTS:

4.1 H/W System Configuration:

- Processor - Pentium –IV
- RAM - 4 GB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - SVGA

4.2 Software Requirements:

- Operating System - Windows XP
- Coding Language - Java/J2EE(JSP,Servlet)
- Front End - J2EE
- Back End – MySQL



5. SYSTEM DESIGN:

5.1 System Architecture:

The system architecture is as follows:

Goals: The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.

2. Provide extendibility and specialization mechanisms to extend the core concepts.

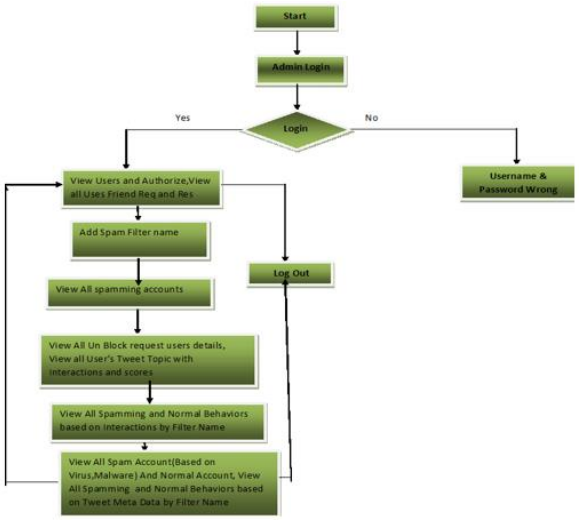


Fig 5.1: System architecture

5.4 Use Case Diagram:

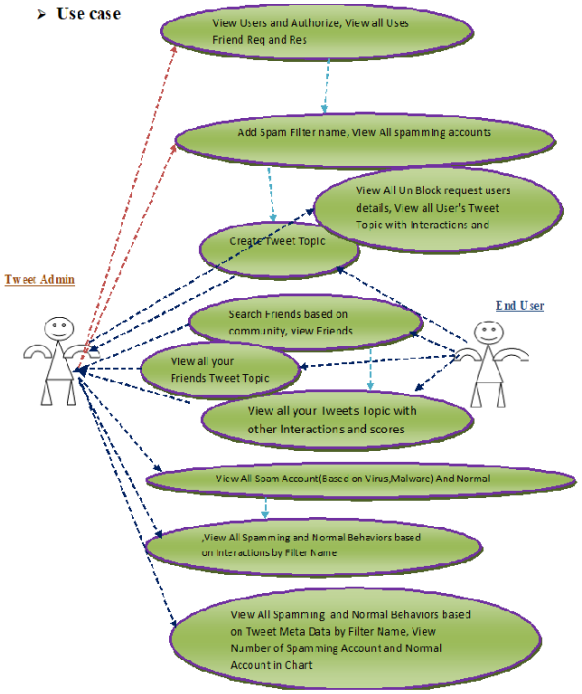


Fig 5.4: Use case Diagram

➢ **Class Diagram :**

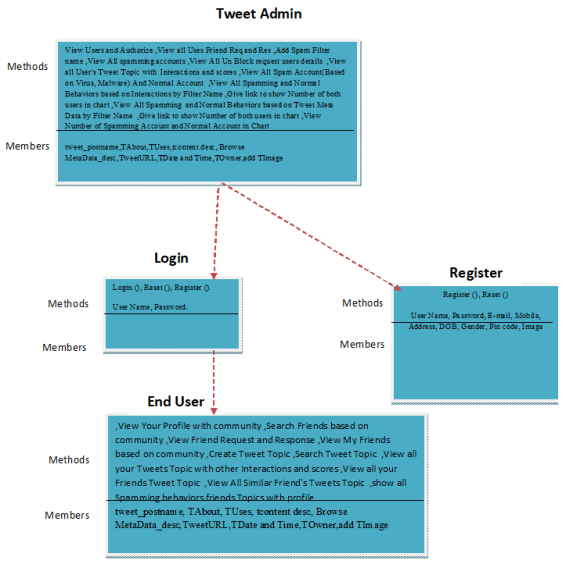


Fig 5.5: Class diagram

5.6 Sequence Diagram:

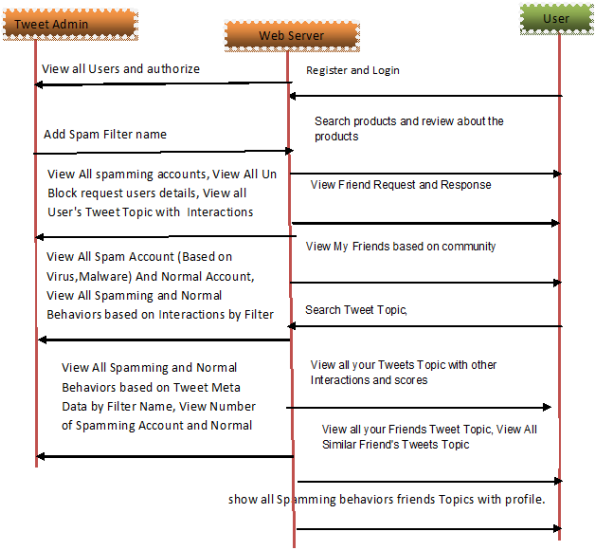


Fig 5.6: Sequence diagram

6. IMPLEMENTATION:

6.1 Sample Code:

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<%@ include file="connect.jsp" %>

<%@ page import="org.bouncycastle.util.encoders.Base64"%>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

```
<title>All User Tweets Page</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style2 {font-size: 18px}
.style3 {
font-size: 24px;
color: #FF0000;
font-weight: bold;
}
.style4 {color: #660033}
.style5 {color:#000000}
.style7 {color:#006633}
.style8 {
font-size: 14px;
font-weight: bold;
}
.style9 {color:#990000}
.style10 {color:#FF6600}
-->
</style>
</head>
<body>
<div class="main">
<div class="header">
<div class="header_resize">
<div class="logo">
<%
String s0="",s1="",s2="",s3="",s4="",s5="",s6="",s7="";
 int i=1,j=0,count=0,s=0;
try
{
String query="select * from tweets";
Statement st=connection.createStatement();
ResultSet rs=st.executeQuery(query);
while ( rs.next() )
{
j=rs.getInt(1);
s1=rs.getString(2);
s2=rs.getString(5);
s3=rs.getString(6);
s4=rs.getString(11);
s5=rs.getString(7);
s6=rs.getString(12);
s7=rs.getString(3);
s=rs.getInt(14);
String decryptedDes= new String(Base64.decode(s5.getBytes()));
count++;
%>
 <tr>
 <td height="0" valign="middle" bgcolor="#FFFFFF">
 <div align="center" class="style5" >
 <div align="center">
 <%out.println(i);%>
 </div>
 </div></td>
<td height="0" valign="middle" bgcolor="#FFFFFF">
<div align="center" class="style9" >
<div align="center">
<strong><%out.println(s0);%></strong>
</div>
</div></td>
<td height="0" valign="middle" bgcolor="#FFFFFF">
```

```
 <div align="center" class="style7" >
 <div align="center">
<strong><%out.println(s7);%></strong>
</div>
</div></td>
Statement st1 = connection.createStatement();
String query1 ="update user set status='"+str+"' where id="+id+" ";
st1.executeUpdate (query1);
connection.close();
response.sendRedirect("Admin_AuthorizeUsers.jsp");
}
catch(Exception e)
{
out.println(e.getMessage());
}
%>
```

Fig 6.1: Home Screen

6.3.2 Admin Page:

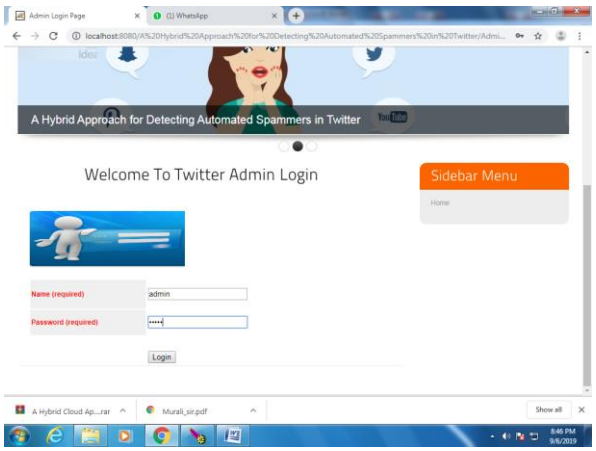


Fig 6.2: Admin Page

6.3.3 Admin Home Page:

Fig 6.3: Admin Home Page

6.3.4 User Profile Page:

Fig 6.4: User Profile Page

6.3.5 Add Spam Filters:

Fig 6.5: Add Spam Filters

6.3.6 View Filters:

Fig 6.6: View Filters

6.3.7 Set Community:

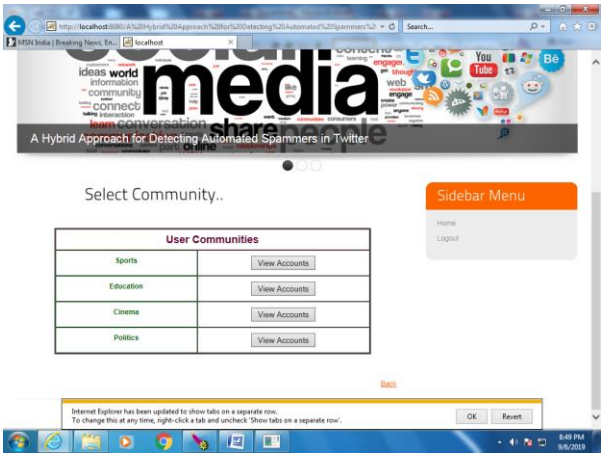


Fig 6.7: Set Community

7. SYSTEM TESTING:

7.1 Introduction:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

7.2 Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields.

7.2.1 Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

- Valid Input: identified classes of valid input must be accepted.
- Invalid Input: identified classes of invalid input must be rejected.
- Functions: identified functions must be exercised.
- Output: identified classes of application outputs must be exercised.

7.3 Other Testing Methodologies:

7.3.1 User Acceptance Testing:

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

8. CONCLUSION:

In this paper, we have proposed a hybrid approach exploiting community-based features with *metadata*-, *content*-, and *interaction-based* features for detecting automated spammers in Twitter. Spammers are generally planted in OSNs for varied purposes, but absence of real-life identity hinders them to join the trust network of benign users. Therefore, spammers randomly follow a number of users, but rarely followed back by them, which results in low edge density among their *followers* and *followings*.

This type of spammers interaction pattern can be exploited for the development of effective spammers detection systems. Unlike existing approaches of characterizing spammers based on their own profiles, the novelty of the proposed approach lies in the characterization of a spammer based on its neighboring nodes (especially, the followers) and their interaction network.

REFERENCES

1. M. Tsikerdekis, “Identity deception prevention using common contribution network data,” *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 1,pp. 188–199, Jan. 2017.
2. T. Anwar and M. Abulaish, “Ranking radically influential Web forum users,” *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 6,pp. 1289–1298, Jun. 2015.
3. Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, “Design and analysis of social botnet,” *Comput. Netw.*, vol. 57, no. 2, pp. 556–578,2013.

4. D. Fletcher, "A brief history of spam," *TIME*, Nov. 2, 2009.[Online]. Available:http://www.time.com/time/business/article/0,8599,1933796,00.html
5. Y. Boshmaf, M. Ripeanu, K. Beznosov, and E. Santos-Neto, "Thwarting fake OSN accounts by predicting their victims," in *Proc. AISec*, Denver,CO, USA, 2015, pp. 81–89.
6. A. A. Amleshwaram, N. Reddy, S. Yadav, G. Gu, andC. Yang, "CATS: Characterizing automation of Twitter spammers," in *Proc. COMSNETS*, Bengaluru, India, Jan. 2013, pp. 1–10.
7. K. Lee, J. C. Lee, and S. Webb, "Uncovering social spammers: Socialhoneypots + machine learning," in *Proc. SIGIR*, Geneva, Switzerland,Jul. 2010, pp. 435–442.
8. G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on socialnetworks," in *Proc. ACSAC*, Austin, TX, USA, 2010, pp. 1–9.
9. H. Yu, M. Kaminsky, P. B. Gibbons, and A. D. Flaxman, "SybilGuard:Defending against sybil attacks via social networks," *IEEE/ACM Trans.Netw.*, vol. 16, no. 3, pp. 576–589, Jun. 2008.
10. H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao, "Detectingand characterizing social spam campaigns," in *Proc. IMC*, Melbourne,VIC, Australia, 2001, pp. 35–47.
11. W. Wei, F. Xu, C. C. Tan, and Q. Li "Sybildefender: Defend against sybil attacks in large social networks," in *Proc. INFOCOM*, Orlando, FL, USA, Mar. 2012, pp. 1951–1959.
12. C. Yang, R. C. Harkreader, and G. Gu, "Die free or live hard? Empirical evaluation and new design for fighting evolving Twitter spammers," in *Proc. RAID*, Menlo Park, CA, USA, 2011, pp. 318–337.