Identification of Suspicious Users on Twitter - A Semantic Approach

Rishab Ketan Doshi - 12IT59
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: rishabketandoshi@gmail.com

Shravan Karthik - 12IT77
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: shravan1994@gmail.com

***Abstract*—**

Social Networking sites such as twitter ensure that information is disseminated to a large audience. Most social networking sites give users the entire freedom to post any sort of content on their respective accounts. This content can be graphical, textual or a web-link provided by the user. Due to the unrestricted nature of the content posted by each user, social networks provide an interesting platform to learn more about each user. This platform can also be used to identify and list suspicious users who post extremist or malicious content. Analysis of a users social network content and behaviour can potentially be used to broadly categorize users into multiple categories and can certainly be used to identify users that tend to be harmful to a society. Identification of suspicious users on a social networking site, can be used by law enforcement authorities to further keep tabs on these users, and can be used to perform other analytics and deductions.

The model proposed in this paper analyses every tweet of the user. A semantic model is proposed which factors in the sentiment of the users content and the system is able to cluster a group of users actively discussing a contentious topic.

I. Introduction

Social networking sites such as the Twitter[1] have roughly 500 million active users. Twitter provides users to send a 140 character message known as tweet. Each user has a set of followers and follows certain users. Each tweet, can comprise of text, website URLs and pictures. Users also have the option to re-tweet' a tweet, which allows a user to endorse a tweet to his followers. Due to largely unregulated nature of the users content users tweet and follow their interests, beliefs. It provides a great opportunity to know more about the users. This also serves as a vital resource to identify suspicious users and activities. While social media and networking sites are generally used to share personal information and tweet on user interests, there exists a lot of users and accounts on social networking sites which spew hatred, extremist views which can be potentially harmful. Due to ease of dissemination of information, content that is harmful is made available to a large audience, which can further lead to social instability. While there exists mechanisms to ensure the removal of these users, most implementation factor in reports and complaints from other users.

The paper proposes a system that automates this process of identifying suspicious users, and providing an effective visualization of the same.

II. Literature Survey

There have been many proposals to help resolve identification of suspicious users, however none of the proposed solutions to the best of our knowledge analyses the sentiment of users tweet. Julei Fu et.al.[2] proposed a six element analysis method to identify suspicious activities based on social network data. However this data model works on historic and static data-set, where communication transcripts were analysed. Skillicorn and David[3] proposed a method to message rank matrices. This method helps filter certain subset of users messages from a cluster of many other messages. Sharath Kumar et.al.[4] also propose a method to identify suspicious users clusters using NLP and latent semantic analysis on a live stream of data but doesn't account for the sentiment analysis of a tweet.We propose a model that factors in the users tweets sentiment and performs a sentiment analysis on the content of the users tweet.

The system factors into account the sentiment of a users tweet along with its content. The content is of the tweet is matched with an exhaustive list of suspicious users. These two parameters are fed into a system which categorises users as suspicious or not. Under the cluster of suspicious users, each user is categorized into broad categories based on the broad topic under which they are found to be suspicious. This monitoring system is fed live tweets and constantly updates and monitors its clusters thereby adding new users to the clusters. A major advantage of this system is that the system adapts with time, thus accurately depicting the cluster to which the user belongs to. Section III in the paper formally defines the problem, Section IV discusses the objectives the system proposed seeks to address. Section V describes in detail the methodology and working of the system. The results of the proposed system are discussed in Section VI and Section VII we present our conclusions.

III. Problem Statement

Given a set of tweets belonging to 'K' distinct users, identify a set of suspicious users and provide a visualization of these suspicious users in the network. In addition these users are clustered to the suspicious topics to which they belong to.

IV. Objectives

The objective of the system is to identify a set of suspicious users and accordingly categorize them into clusters. Another round of clustering identifies the broad category of suspicious activity to which the user belongs to and finally a visualization of these sets is performed to better understand the nature of relationships between users.

V. Methodology and Work Done

The work flow of the proposed system is as follows :-

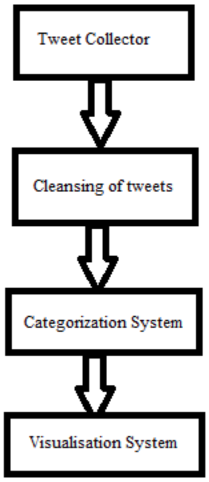


Figure 1: High Level Flow Chart

- Tweet Collector
- Cleaning of tweets
- Categorization System
- Visualization of clusters

A server is set to assemble live tweets. These tweets are assembled using the standard twitter rest API. The collected tweet is passed through a data cleansing station which performs the following functions:

1. Extraction of the text of the tweet.
2. Removal of stop words.
3. Aggregating of tweet to users feature list

```
while tweets remain do
    Extract Text, UserId of tweet;
    Remove hashtags from Text;
    Remove stop words from Text;
    Remove Mentions from Text;
    Remove twitter specific keywords from Text;
    Append to feature-list[UserId] filtered Text;
end
```

Algorithm 1: Pseudo code for building of feature list:

A feature list is an aggregation of all the tweets associated with a particular user. This feature list is further fed to the next step which involves semantically analysing every tweet of the user. In the semantic model, an exhaustive list of suspicious words is built. Each tweet of the user is analysed and is assigned a score which is normalized, indicating the nature of the tweet.

```
while users are left do
    SentiScore[uId]=0;
    SuspiciousMatch[userId]=0;
    while tweets in feature-list[uId] do
        SentimentScore=getSentimentScore(tweet);
        SentiScore[uId]=SentiScore[uId]+SentimentScore;
        while words in SuspiciousBag do
            SuspMatch[uId]=SuspMatch[uId]+fuzzWuzz(word,tweet);

        end
    end
    SentiScore[userId]=SentiScore[userId]/len(feature-
    list[userId]);

    SuspiciousMatch[userId] =
    SuspiciousMatch[userId]/len(feature-list[userId]);
end
```

Algorithm 2: Pseudo code for Clustering:

Sentiment analysis is performed next on the tweet. The tweet has various sentiments or tones to which it can be associated with. Some of these include positive, negative, sarcasm etc. A user with a higher score with the semantic model, and has a high value for negative sentiment, is most likely to be a suspicious users. The reason for inclusion of sentiment analysis of users is that while the users tweet may largely contain words consistent with the suspicious list this, the entire tone of sentiment of the tweet may be that of sarcasm and critical of the activities classified as suspicious. This indicates a benign user and without the presence of sentiment analysis the users would be largely classified as suspicious. Another advantage of the sentiment analysis is that while there may be some words not present in the suspicious list, a constant negative sentiment of the tweet indicates a suspicious behaviour of the user. The sentiment analysis for each tweet is performed using AlchemyAPI, which takes in the users tweet and returns a sentiment score of the user. The sentiment score comprises of two portions - the sentiment score of the tweet as a block and the sentiment score associated with the keywords in the tweet.

The list of suspicious words is obtained using an exhaustive list prepared by the Department of Homeland Security(DHS). This list is compared with the feature-list associated with a given user and the suspicious word list count is calculated. An in-built python library called FuzzyWuzzy is used to perform string comparison. An interesting feature of this system is that this list is dynamic, i.e the initial data-set used is the list from the DHS however once a user is identified as being suspicious keywords with strong negative sentiment which didn't exist in the suspicious keywords list is appended. This ensures greater accuracy of the system.

The SentiScore and SuspiciousMatch vector is fed to various learning algorithms. The SentiScore keeps track of two important parameters. The block sentiment score, which features the sentiment score of the entire tweet and the keyword sentiment score which factors in the sentiment of the keywords associated with the tweet. As demonstrated in the results these two play a vital role in determining if a user is suspicious or not. Another important factor is the count of the suspicious words which is obtained from SuspMatch, which indicates the number of suspicious words associated with each user.

The SentiScore and SuspiciousMatch values for each user are normalised. Ensemble methods such as Random forest algorithm are used, which factors in multiple attributes and builds the decision tree, and categorises users as either suspicious or not. The list of suspicious users is then fed to a clustering algorithm, that clusters the suspicious users based the broad topic under which they are suspicious. For ensuring an accurate model, the training data-set is further split into training and cross-validation set.

The output from these learning algorithms is then fed into a visualization tool that firstly classifies users as benign or suspicious. Later the visualization tool clusters users belonging to their respective suspicious category subject.

VI. Results and Analysis

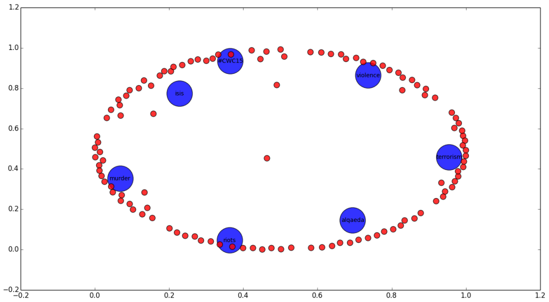


Figure 2: Different Topics from which tweets were pulled

A collection of many users spanned across the social network was considered for the training set. Accounts linked with terrorist organizations were used for the training model and an exhaustive list of words associated with these accounts was used to semantically model the category to which the user belongs to. Once the training data-set was established the test random forest algorithm was trained by splitting the training data-set further into training data-set and cross validation set. Once the system has been trained users tweets was fed into the users list. The test set was fed through the system and categorises the user as suspicious or not suspicious. The suspicious users are later appended to the training set, and the accuracy of the users classification increases. From the list of suspicious users a visualization of the clusters to which these suspicious users belong to is built. This comprises widely of the list of topics that are pertinent to suspicious activity.

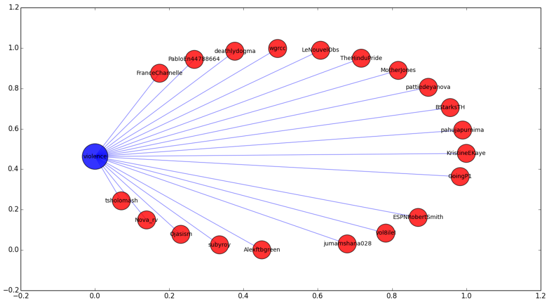


Figure 3: Users related to the keyword Violence

The system also factors into account the change in the behavior of the user. Thus updating the category to which the user belongs to. This dynamic model helps detect live changes in the users behavior and once a user has been detected as suspicious, the user is payed closer attention to closely scrutinise the users behavior.

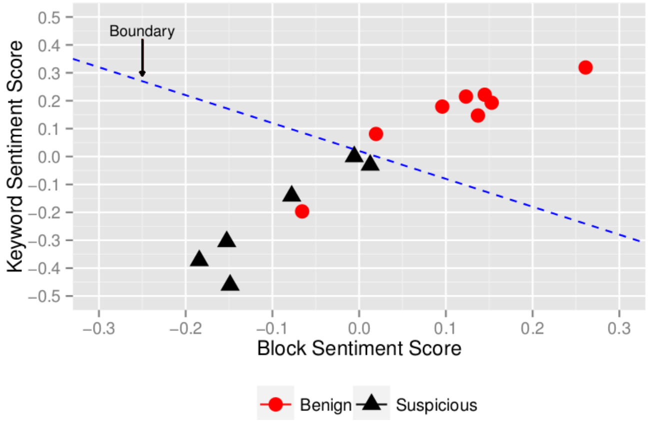


Figure 4 : Clusters of suspicious and benign users

The accuracy of the algorithm is assessed using precision and recall values.

The confusion matrix for the categorization of users is : Precision and recall for the confusion matrix are as below

TABLE I
Confusion Matrix for Classification of suspicious users using Random Forest Algorithm

-	True	False
True	16	1
False	1	0

1) Precision = 16/(16+1) = 0.941
2) Recall = 16/(16+1) = 0.941

As indicated the system has a high percentage of values it classifies correctly which corresponds to the high precision of the algorithm.

Clustering of users is also done using the K-Nearest Neighbors algorithm. However as results are shown below the results suffer from a high bias and thus a low precision is obtained.

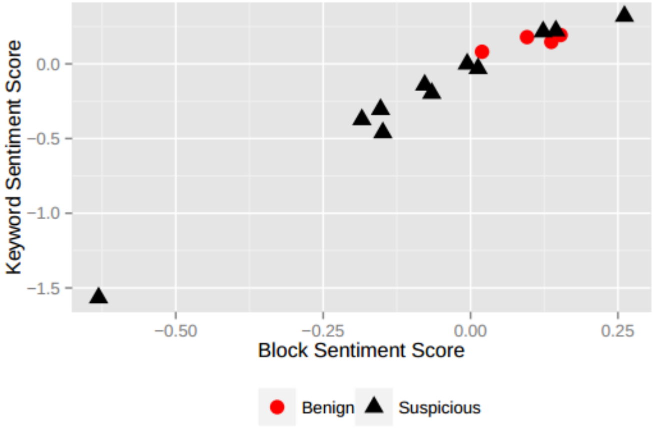


Figure 5: Classification using K Nearest Negihbors Algorithm

TABLE II
CONFUSION MATRIX FOR CLASSIFICATION OF SUSPICIOUS USERS USING K-NEAREST NEIGHBORS

-	True	False
True	13	4
False	1	0

The Precision and recall values for the above confusion matrix are as below

1) Precision = 13/(13+4) = 0.764
2) Recall = 13/(13+1) = 0.928

The precision value is far less for K-Nearest Neighbors algorithm. This is because the clustering algorithm accounts for the euclidean distance between points. This is a naive way to cluster points given the ambiguous nature of parameters considered.

The utilization of ensemble methods such as random forest which builds a decision tree and picks the decision tree with least entropy at each level helps increase the accuracy of the classification of the user. This approach has greater accuracy than the clustering algorithm, or the naive algorithm suggested in the base paper where a threshold is selected for a parameter. If the value of a parameter exceeds a threshold the user is classified as suspicious.

The visualization tool also performs a great way for the law enforcement agencies and others involved to identify list of suspicious user. The visualization tool clearly defines a network of suspicious users with clear distinctions and connections between varied users. This can be used to identify users that are related to suspicious users, and thus may not be recognized by the system to be suspicious, however can be monitored to observe their social network behavior.

The output of the random forest algorithm is fed to a R script, which uses in-built 'ggplot2' package to perform visualization. A plot comprising of the block sentiment score versus the keywords sentiment score is built, and using a linear regression model a boundary between the two clusters is drawn. As indicated in the given data-set there exists one anomaly - outlier which affects the precision of the system.

Using the python package NetworkX, clusters of users are depicted along with a list of suspicious words associated with each user. This helps provide a better understanding of why a user is classified suspicious and list of all users associated with a particular topic.

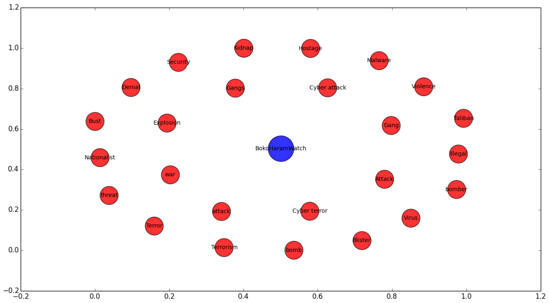


Figure 6: Suspicious Words associated with BokoHaramWatch

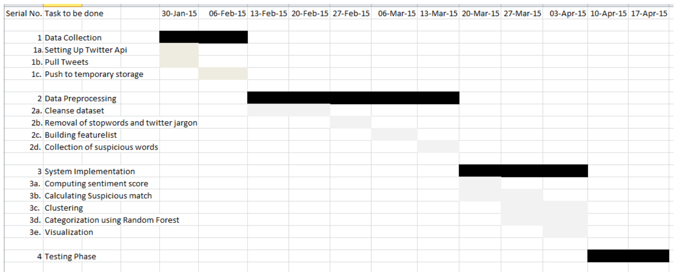
VII. CONCLUSION

While social networks are increasingly used as a medium of dissemination of information, it also serves as a place to spread extremist views. Thus incorporating the above system would help identify these users and block their accounts, to ensure that the society as a whole is not harmed.

ACKNOWLEDGMENT

The authors would like to thank Prof. GRM Reddy, Mr.Ashwin Dixit and Mr. Shridhar Domanal for their valuable guidance.

INDIVIDUAL CONTRIBUTION



Serial No.	Task to be done	30-Jan-15	06-Feb-15	13-Feb-15	20-Feb-15	27-Feb-15	06-Mar-15	13-Mar-15	20-Mar-15	27-Mar-15	03-Apr-15	10-Apr-15	17-Apr-15
1	Data Collection	■	■										
1a.	Setting Up Twitter Api	▨											
1b.	Pull Tweets	▨											
1c.	Push to temporary storage		▨										
2	Data Preprocessing			■	■	■	■	■					
2a.	Cleanse dataset			▨	▨								
2b.	Removal of stopwords and twitter jargon					▨							
2c.	Building featurelist						▨						
2d.	Collection of suspicious words							▨					
3	System Implementation								■	■	■		
3a.	Computing sentiment score								▨				
3b.	Calculating Suspicious match								▨	▨			
3c.	Clustering									▨	▨		
3d.	Categorization using Random Forest									▨	▨		
3e.	Visualization										▨		
4	Testing Phase											■	■

Figure 5: Gantt Chart

1) Rishab Ketan Doshi
    - Pull Tweets
    - Cleanse dataset
    - Push to temporary storage
    - Building featurelist
    - Collection of suspicious words
    - Computing sentiment score
2) Shravan Karthik
    - Calculating Suspicious match

- Clustering
- Categorization using Random Forest
- Visualization

Implemented/Base Paper

Sharath Kumar Sanjay Singh, ”Detection of User Cluster with Suspicious Activity in Online Social Networking Sites” Advanced Computing, Networking and Security (ADCONS), 2013 2nd International Conference 15-17 Dec 2013. pp. 220-225

References

[1] https://www.twitter.com

[2] F. J. Fu, J. Chai, and S. Wangl., Multi-factor analysis of terrorist activities based on social network, Business Intelligence and Financial Engineering (BIFE), 2012 Fifth International Conference on 18-21 Aug. 2012, pp. 476480, 2012.

[3] Skillicorn and David, Keyword ltering for message and conversation detection, Queens University.[Available Online] http://www.cs.queensu.ca/home/skill/beyondkeywords.pdf, 2005.

[4] Sharath Kumar Sanjay Singh, ”Detection of User Cluster with Suspicious Activity in Online Social Networking Sites” Advanced Computing, Networking and Security (ADCONS), 2013 2nd International Conference 15-17 Dec 2013. pp. 220-225

[5] Fuzzywuzzy Library for string matching https://github.com/seatgeek/fuzzywuzzy

[6] NLTK package-Semantic modelling of tweets in python.

[7] AlchemyApi-For sentiment analysis of tweets http://www.alchemyapi.com/

[8] DHS List of suspicious words. https://gist.github.com/jm3/2815378

[9] NetworkX- Visualization of Clusters https://networkx.github.io/

[10] GGPlot2- Interactive plot for visualization http://ggplot2.org/