iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** VI **Month of publication:** June 2019

DOI: http://doi.org/10.22214/ijraset.2019.6314

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue VI, June 2019- Available at www.ijraset.com

Network Analysis of Recurring Twitter Spam Campaign

Akhil Dixit[1], Arush Agarwal[2]
[1]*Electronics and Communication, Delhi Technological University, Bawana, Delhi*
[2]*Electronics and Communication, Netaji Subhas University of Technology, Dwarka, Delhi*

Abstract: ***Online Social Networks (OSNs) such as Twitter and Facebook have become popular communication and information sharing tools for hundreds of millions of individuals in recent years. OSNs not only make peoples life more connected, but also attract the interest of spammers. Twitter spam generally contains deceptive information, such as free voucher and weight loss advertisement to attract the interest of victims. A comprehensive analysis on the deceptive information will be of great benefit to the detection of Twitter spam.***

Twitter has now become one of the largest online social network sites. Over 500 million registered users spend vast time making friends with people who they are familiar with or interested in. For Twitter users, after relationships are built, they can receive tweets, usually something interesting or recent activities shared by their friends. Nowadays, Twitter has largely shortened the distance of people, and reshaped the way they communicate with each other.

Current security experts suggest the best defense against spam is to educate Internet users to never click suspicious links in tweets. In the real world, however, the effectiveness of education is far from our expectations. Spammers leverage some certain deceptive topics, such as gain followers, cracked games, i.e. to lure users to click their malicious links. We refer this kind of information as deceptive information.

The deceptive information is one of the key factors to the spreading efficiency of spam on Twitter. A better understanding of deceptive information is crucial to spam detection techniques. Therefore, we are motivated to thoroughly study the deceptive information employed by spammers.

Keywords: Twitter; Spam Analysis; NetworkX; Jaccard Distance; FANMOD Analysis

I. INTRODUCTION

Cybercriminals have leveraged the popularity of a large user base available on Online Social Networks (OSNs) to spread spam campaigns by propagating phishing URLs, attaching malicious contents,etc.

Twitter is the social media platform which provide a social network of over 500 million registered users post messages known as tweet. As Twitters reach expands, it has also become more attractive to spammers. Twitter spam is referred as unsolicited tweets that lure users to malicious sites containing malware downloads, scams etc. Twitter relies on blacklists to block spam, but this filtering only suppresses spam links that are blacklisted at the time of posting.

Moreover, most spam was posted on Twitter in the form of short URLs. Spammers leverage some certain deceptive topics, such as gain followers, free items, i.e. to lure users to click their malicious links. This kind of information is referred as Deceptive Information.

This lets spammers create a network among the malicious tweets and users contributing those tweets. This led us to pursue this project in order to identify such recurring spam campaigns/networks and analyze them in order to minimize spamming scandals on Twitter.

We collected tweets containing particular hashtags from Twitter based on an exhaustive list of 20 keywords via Twitter streaming API. We chose Twitter due to easy availability of data. We model the Twitter dataset as a heterogeneous network by leveraging various interconnections between different types of nodes present in the dataset.

Our research has found that bot-posted spam tweets are often associated with orchestrated campaigns that can remain active for long periods of time, where the primary targets are trending topics like free vouchers,weight loss tips,food discounts, etc that easily attract users[1]. This research paper presents an evaluation of the detection of these recurring campaigns using network analysis, based on networks derived from the tweets posted by users.

Finally, we perform a case study to show how our method is capable of detecting those users as spammers who have not been suspended by Twitter (and other baselines) yet.

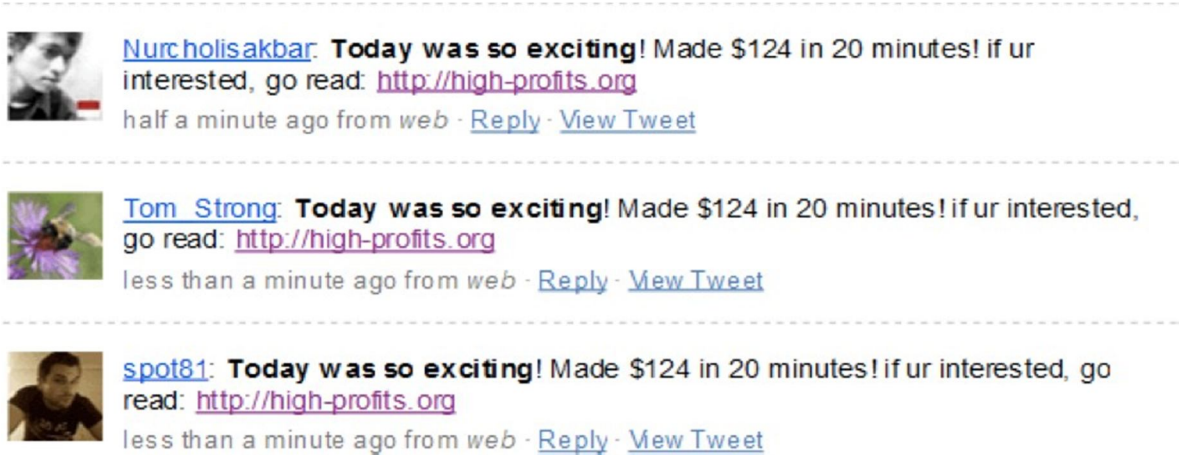


Fig. 1 An example of Twitter Spam

II. AVAILABLE LITERATURE REVIEW

In order to better understand Twitter spam, some in-depth analysis has been carried out.

Grier et al. analysed 25 million URLs from 200 million public tweets, and found that 2 million URLs were spam, which accounts for 8% of all crawled unique URLs experimented by Benevenuto and F. Rodrigues[2] in "Detecting Spammers and Content Promoters in online video social networks".

The decision trees algorithm used by Milo, R. Itzkovitz experimented "Superfamilies of Evolved and Designed Networks"[3] to identify link based web-spams. The research recognized and eliminated links between pages which were present for reasons other than merit using heuristics to drop internal links. Drost et. al. [3] used SVM to classify web spam with content based features.

D. Metzler, S. Dumais, and C. Meek[4] in their research work on "Similarity Measures for Short Segments of Text" followed the research work of Chao Chena, Sheng Wena , Jun Zhanga[8] given in their paper on "Investigating the deceptive information in Twitter Spam".

In [5] Mishne et. al. used machine learning techniques and probability distributions over strings for the detection of spam comments. They compare the language models used in the blog post, in the comment and in the pages linked by the comments. In [8] Bhattari et. al. used the data corpus from the same set to detect spam comments from blogs and sites.

Jenq-Haur Wang and Ming-Sheng Lin detected spam comments on Chinese blogs using the concept of inter-comment similar- ity, which is the similarity among all comments for the same blog post[9]. In order to estimate the similarity between comments, they used measures such as Jaccard and Dice coefficients.

They concluded that Jaccard similarity for post-comment and inter-comment similarity scored together with non-content features such as stopwords ratio and link number achieved the best performance under C4.5 classification (in terms of F-measure). A survey by Srishti Gupta et. al.[10] showed that the three main anti-spam strategies commonly used in practice are:

Identication-based, Rank-based and Interface or Limit-based. It mentioned that the third method has been used to prevent comment spam.

K. Thomas et. al.[11] has reviewed the current state of spam in the blogosphere, concluding that anti-spam solutions devel- oped for emails are effective in detecting blog.

A.H. Wang[12] has considered the problem of content-based spam filtering that arises in three contexts, blog comments in- volved. Their experiments are conducted to evaluate the effectiveness of state-of-the-art email spam filters, without modification. Further experiments are conducted to investigate the effect of lexical feature expansion. Detection of Harassment on Web2.0

C. Yang [13] employs content features, sentiment features and contextual features of documents, using a supervised learning approach to identify online harassment, including comment spam.

Regarding other online social networking websites, Gao et al. investigated spam within Facebook wall messages, using networks based on message similarity. The shortcomings of URL blacklists for Twitter spam prevention were highlighted by Grier et al., with related analysis of shortened URLs by Chhabra et al.[14].

Derek OCallaghan, Martin Harrigan, Joe Carthy and P adraig Cunningham of the School of Computer Science Informatics,

University College Dublin presented an evaluation of the detection of recurring spam campaigns using network analysis, based on networks derived from the comments posted by users to videos[15].

The network was created using the concept of similarity, taking into consideration the Jaccard distance between various comments[16]. Short text similarity estimation has attracted attention in Web applications. Huanget. al [17] proposed to build hierarchical classifiers from short text segments using Web search-result snippets as corpora. Sahami and Heilman [18] proposed a Web-based kernel function for measuring the similarity of short text snippets. Metzler, Dumais, and Meek [19] addressed various ways of text representation including content, stemmed, and expanded representation using contextual information from external sources. In this paper, we simply estimate the similarity between blog posts and comments by calculating the overlap in word n-grams. In relation to analysis of spam within YouTube, Benevenuto et al. created a directed network based on videos and associated responses, and used derived features as part of a larger set for spam user detection. Separate spam investigations include that of Sureka [20].

Metzler, Dumais, and Meek addressed various ways of text representation including content, stemmed, and expanded representation using contextual information from external sources[21].

Currently researchers are using two ways to label spam, manual inspection and blacklists filtering, e.g. google safebrowsing. While manual inspection can label a small amount of training data, it is very time- and resource-consuming. A large group of people is needed to help during the process. Although HIT (human intelligence task) websites can help to label the tweets, it is also costly and sometimes the results are doubtful . Others apply existing blacklisting service, such as Google SafeBrowsing to label spam tweets. Nevertheless, these services API limits make it impossible to label a large amount of tweets.

III. WORK PLAN

In this research work, we do analysis of Twitter Spam Campaign. The proposed algorithm parses the tweets and finds the similarities of similar texts and recurring messages having short period of origin source. This infers that particular message is being transmitted over multiple twitter handles, having the same text and similar time of origin which predicts that a computer bot is spamming across the channel. The algorithm then filter out the data on the aspects of keyword that are deceptive in nature. It then analyses the data to anticipate spam tweet and associated handle. The algorithm analyses in the following steps:

A. Gather Data

Our first step was to fetch data from Twitter API which consists of millions of tweets posted over the last few weeks. Initially we collected the data of 2 days which comprised of 63915 tweets.

B. Filter Data

Then, our next step was to filter out tweets based on various deceptive topics such as weight loss advertisement, free vouchers, ’deals’, ’promos’ etc that promote spams and lure users to visit such links[8]. We chose to analyze tweets of such topics due to the interest of users in such areas.

C. Identify Potential Spam Tweets

Based on our research, we then analyzed that possible spam tweets consist of large number of URLs, hashtags and mentions. This is because a spam user tend to spread the network by including malicious website links or mentions of malicious user proles. This contributes to spreading spam campaigns since users of twitter tend to click on such illegal website URLs unknowingly.

This led us to identifying and separating potential spam tweets from legitimate tweets.

D. Analyzing Tweets of each Deceptive Topic

On the basis of our research, we initially selected 5 deceptive topics where we could find potential spam tweets. The categories were as follows:-

1) Free coupon
2) Giftvoucher
3) Deals
4) Giveaway
5) Promo

E. Segregate Actual Spam Tweets

Thereafter, we analyzed 200 tweets consisting of URLs of each deceptive category manually through the technique of crowd-sourcing. From the 200 analyzed tweets, we tried to obtain actual spam tweets.

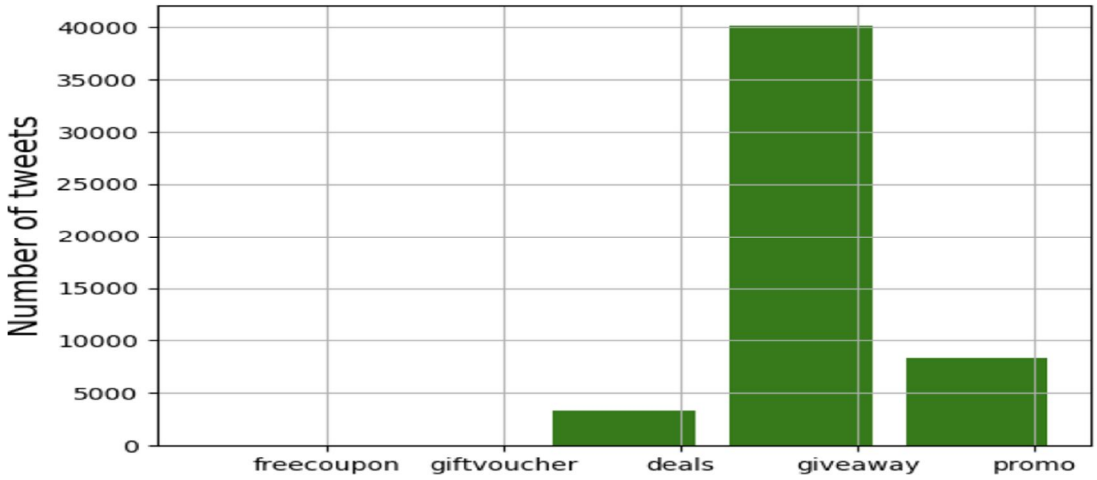


Fig. 2 Distribution of Tweets

Spam tweets could be obtained by performing 2 steps:-

1) We compared the link present in each tweet with a database of malicious URLs. If we found a match, then we flagged the respective tweets as a spam tweet since it consisted of malicious URL.

2) In case we didn't a match, we observed whether the URL present, is in context with the tweet or not. If yes, we ignored it. Else we marked it that tweet as a spam tweet.

F. Create Spam Network

Once a spam tweet has been obtained, we analyzed if there were other users who were involved in tweeting the same tweet or a similar tweet[22]. Similarity of tweets is checked based on the method of Jaccard Similarity Distance.

Jaccard Distance between 2 strings(str1, str2) is computed as:- a = set(str1.split())

b = set(str2.split()) c = a.intersection(b)

return (1- oat(len(c)) /len(a|b))

Here the value of Jaccard Distance value lies between 0.0 and 1.0. If the value is 0.0, then the 2 strings turn out to be highly similar(or same). If the value is 1.0, then the 2 strings are highly dissimilar(or totally different).

Once we have found out all the similar tweets, a graph is created thereafter. Nodes of the graph= Users of the similar/same tweets An edge is created between 2 nodes if the value of Jaccard Distance between their respective tweets is less than 0.4(thresh- old value).

The flow chart of the following methodology is depicted below:-

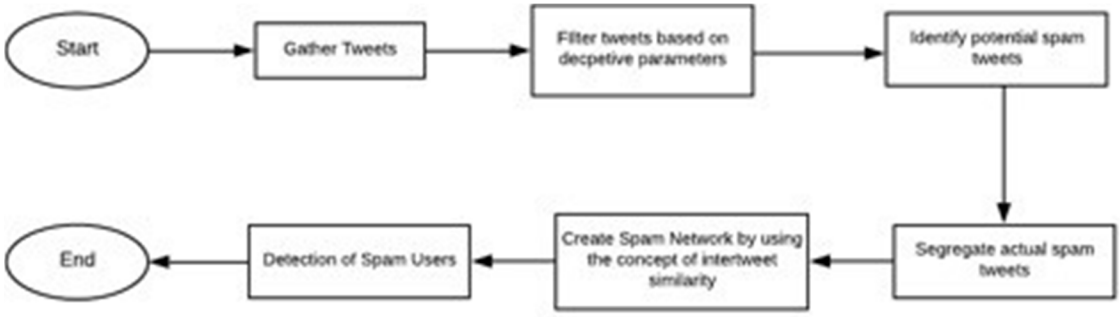


Fig. 3 Data Flow Diagram

IV. IMPLEMENTATION DETAILS

The code of the following project is written under python language along with the incorporation python packages such as pandas, numbpy etc. The code snippets are as follows:-

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue VI, June 2019- Available at www.ijraset.com

V. RESULTS AND ANALYSIS

A. *Results*

1) Total number of tweets extracted in 2 days are 63915.
2) The spam campaign obtained lies under the category of giveaway.
3) Total number of tweets extracted under the category giveaway are 8039.
4) There are total 45 users involved in spreading the single spam campaign under consideration.
5) The jaccard distance between every pair of spam tweets is 0.0 which proves that tweets are highly similar(or same in this case).

File Edit Selection Find View Goto Tools Project Preferences Help

cleandata.py | cleandata_graph.py | Searching.2018.720p.BluRay.x264-DRONES.srt

```python
#!python2
import json
import pandas as pd
# import matplotlib.pyplot as plt
import re
from collections import defaultdict

tweets_data_path = 'collect_data.txt'

tweets_data = []
tweets_file = open(tweets_data_path, "r")
for line in tweets_file:
    try:
        tweet = json.loads(line)
        tweets_data.append(tweet)
    except:
        continue
print("Total no. of tweets extracted:")
print (len(tweets_data))
# print tweets_data.columns

tweets = pd.DataFrame()
tweets['text'] = map(lambda tweet: tweet['text'], tweets_data)
tweets['lang'] = map(lambda tweet: tweet['lang'], tweets_data)
tweets['country'] = map(lambda tweet: tweet['place']['country'] if tweet['place'] != None else None, tweets_data)
tweets['user'] = map(lambda tweet: tweet['user'], tweets_data)
tweets['user_name'] = map(lambda tweet: tweet['user']['screen_name'], tweets_data)
tweets['user_id'] = map(lambda tweet: tweet['user']['id'], tweets_data)
tweets['user_name'] = map(lambda tweet: tweet['user']['name'], tweets_data)
tweets['retweet_count'] = map(lambda tweet: tweet['retweet_count'], tweets_data)
tweets['id'] = map(lambda tweet: tweet['id'], tweets_data)
```

Fig. 4 Snippet 1

```python
giveaway=pd.DataFrame(columns=tweets_with_link.columns)
cond2=tweets_with_link['giveaway']==True
rows2=tweets_with_link.loc[cond2,:]
# rows_giveaway=rows2[0:200]
giveaway=giveaway.append(rows2)
print "Total no. of giveaway tweets extracted:"
print len(giveaway)
# print giveaway

def get_spamlist():
    main_link="https://t.co/Qwai8bYu9U"
    spam_list=[]
    for index, row in giveaway.iterrows():
        # print row['link']
        check=row['link']==main_link
        if check:
            spam_list.append(row)
    return spam_list


def get_jaccard_sim(str1, str2):
    a = set(str1.split())
    b = set(str2.split())
    c = a.intersection(b)
    return (1- float(len(c)) /len(a|b))
```

Fig. 5 Snippet 2

Following are various outputs obtained

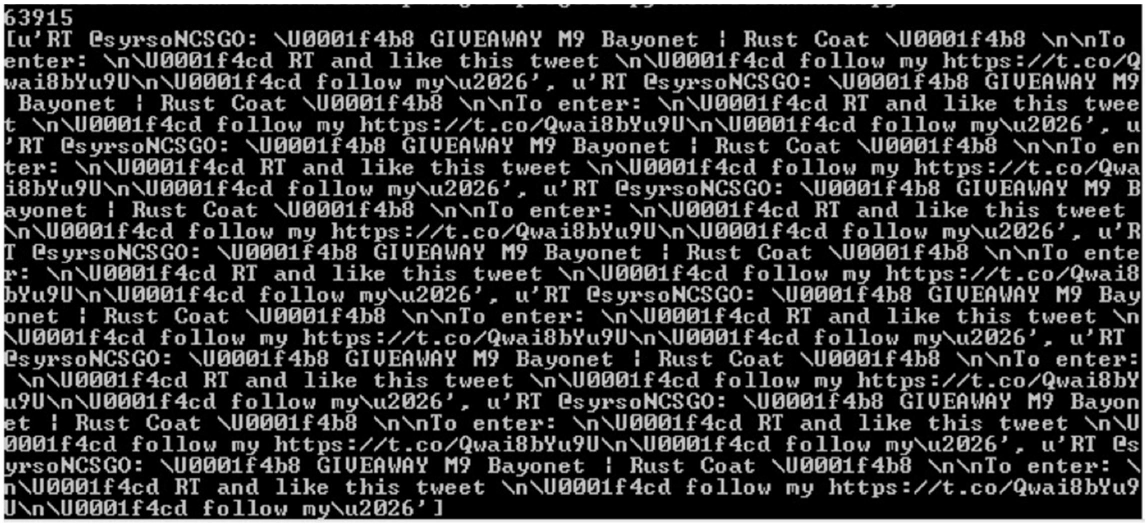


Fig. 6 Figure depicting text of similar tweets

Fig. 7 Distance between initial 9 pairs of spam tweets

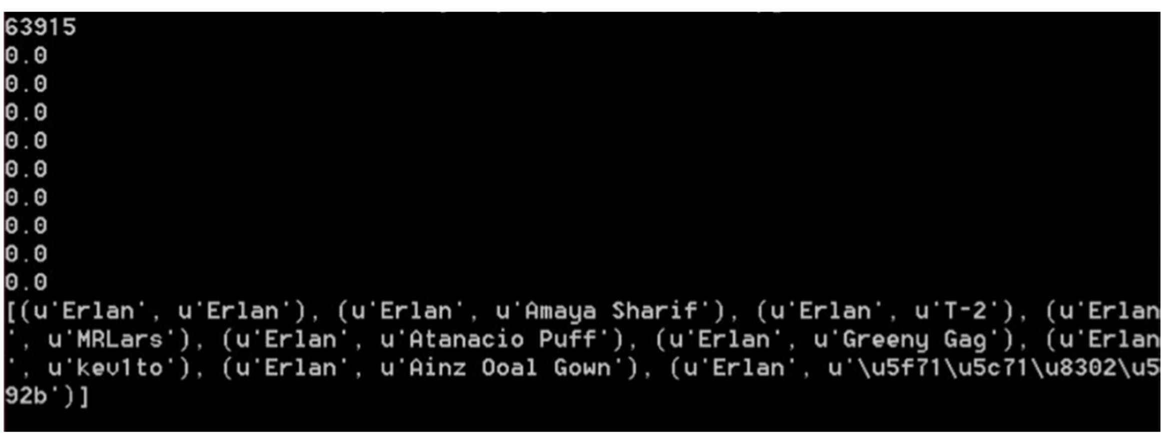


Fig. 8 Depicting edges between dierent pairs of users

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue VI, June 2019- Available at www.ijraset.com

```
Total no. of tweets extracted:
63915
Total no. of giveaway tweets extracted:
8039
Total no. of users involved in spam campaign:
45
Jaccard distance between every pair of users:
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
```

Fig. 9 Depicting total no. of spam users

```
0.0
0.0
0.0
0.0
[(u'Erlan', u'Erlan'), (u'Erlan', u'Amaya Sharif'), (u'Erlan', u'T-2'), (u'Erlan
', u'MRLars'), (u'Erlan', u'Atanacio Puff'), (u'Erlan', u'Greeny Gag'), (u'Erlan
', u'kev1to'), (u'Erlan', u'Ainz Ooal Gown'), (u'Erlan', u'\u5f71\u5c71\u8302\u5
92b'), (u'Erlan', u'QRCatMono'), (u'Erlan', u'Flusha52'), (u'Erlan', u'Ozzie'),
(u'Erlan', u'Lorde'), (u'Erlan', u'Thom'), (u'Erlan', u'@IkerAE gamdom.com idle-
empire.com'), (u'Erlan', u'Master Guardia'), (u'Erlan', u'Norbert Kobos'), (u'Er
lan', u'Tupolew'), (u'Erlan', u'alexp3311'), (u'Erlan', u'Avatar on Speed'), (u'
Erlan', u'Durkki'), (u'Erlan', u'Florian B\xfchrer'), (u'Erlan', u'Youtube'), (u
'Erlan', u'vilonte'), (u'Erlan', u'Guilherme Rodrigues #SkinUP'), (u'Erlan', u'Z
alvo'), (u'Erlan', u'Noah Bertel Haugaard Jensen'), (u'Erlan', u'PLACEBO\u2122')
, (u'Erlan', u'Moritz Ma\xdfbaum'), (u'Erlan', u'YeuNTC'), (u'Erlan', u'777'), (
u'Erlan', u'\U0001f9d4\U0001f3fbNando Pesa'), (u'Erlan', u'Hein Htet Aung'), (u'
Erlan', u'Soldado'), (u'Erlan', u'Marec "Marec" H'), (u'Erlan', u'ihab nemouchi'
), (u'Erlan', u'Gilzeria \U0001f340'), (u'Erlan', u'Anamika'), (u'Erlan', u'3Log
y UnKnown'), (u'Erlan', u'Alec'), (u'Erlan', u'Leszek'), (u'Erlan', u'crsy'), (u
'Erlan', u'Kr0n0z'), (u'Erlan', u'KG20'), (u'Erlan', u'Mathis Klever')]
```

Fig. 10 Depicting information of spam users

B. Data Analysis

We analyzed information of 9 users out of 45 and observed the time intervals at which the spam tweets were posted. We observed that the tweets were spammed at a difference **of as low** as 5 seconds. Information about the 9 users which include user name, time of posting, date of creation, text of tweet etc is given in table1

VI. NETWORK VISUALIZATION

Once our network was obtained, we visualized the network by means of a visualization tool-NetworkX. NetworkX is not primarily a graph drawing package but basic drawing with Matplotlib as well as an interface to use the open source Graphviz software package are included. These are part of the networkx.drawing package and will be imported if possible. It useful to interactively test code using ipython -pylab, which combines the power of ipython and matplotlib and provides a convenient interactive mode.

TABLE I
Details Of A Spam Message Sent To 9 Users

Tweet Time Id	Text
User 1 Sun, Mar 17 10:29:16 +0000 2019	1107227393226211 30"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 2 Sun, Mar 17 10:29:35 +0000 2019	1107227471840071 70"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 3 Sun, Mar 17 10:30:02 +0000 2019	1107227587384737 80"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 4 Sun, Mar 17 10:30:08 +0000 2019	1107227613339312 10"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 5 Sun, Mar 17 10:30:26 +0000 2019	1107227685950939 10"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 6 Sun, Mar 17 10:30:51 +0000 2019	1107227790837841 90"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 7 Sun, Mar 17 10:31:35 +0000 2019	1107227978172383 20"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 8 Sun, Mar 17 10:32:28 +0000 2019	1107228198922653 70"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."
User 9 Sun, Mar 17 10:34:55 +0000 2019	1107228814050943 00"0RT @sysroNCSGO: GIVEAWAY M9 Bayonet | Rust Coat ToEnter: RT and like this tweet follow my https://t.co/Qwai8bYu 9u follow my.."

VII. NETWORK ANALYSIS

Once our graph of spam users is obtained, we aim to analyze the network motifs created in our graph. Each network motif corresponds to a single spam campaign.

Various tools are present to analyze a network motifs. One such tool is FANMOD. FANMOD [7] is a tool for finding so- called networks motifs in a network, that is, it finds small vertex-induced subgraphs that occur significantly more often than in random networks.

For a general introduction to the concept of network motifs, see [23,6]. Fanmod is able to search for network motifs of size between three and eight vertices. Fanmod features a graphical interface for easy setup of algorithm parameters; the results can be exported to HTML.

Following results were obtained by FANMODanalysis:- Network type: Undirected

Number of nodes: 45 Number of edges: 44 Number of single edges: 0

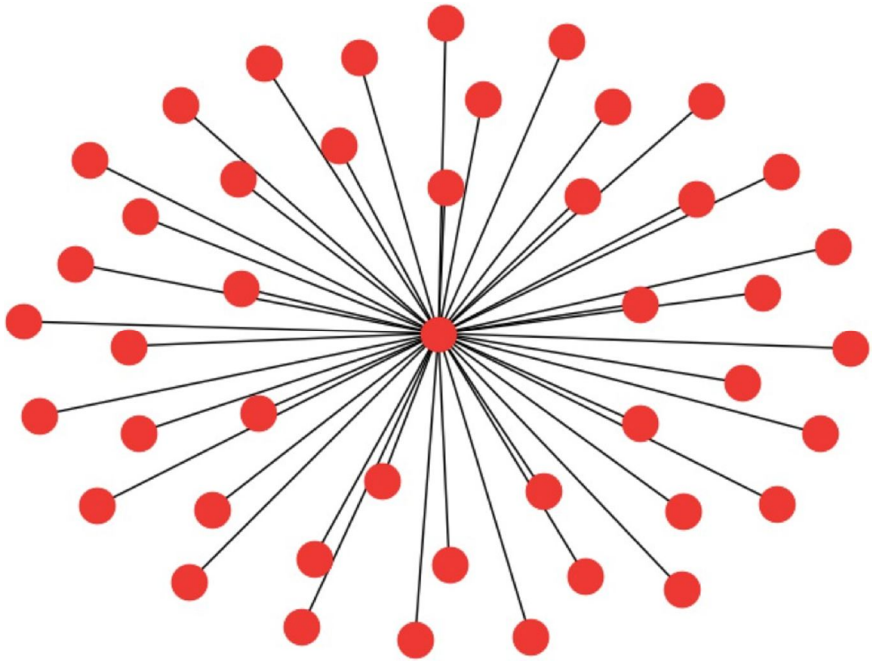


Fig. 11 Network visualization using NetworkX

Number of mutual edges: 44 Algorithm: enumeration Subgraph size: 5

Generated 100 random networks with locally constant number of bidirectional edges, 3 exchanges per edge and 3 tries per edge.

135751 subgraphs were enumerated in the original network. 13575100 subgraphs were enumerated in the random networks.

13710851 subgraphs were enumerated in all networks.

Randomization took 0.003 seconds. Enumeration took 21.058 seconds.

Visualization of the results obtained through FANMOD is tabulated as:-

ID	Adj	Frequency [Original]	Mean-Freq [Random]	Standard-Dev [Random]	Z-Score	p-Value
1082430		100%	100%	0	undefined	0

Fig. 12 FANMOD Analysis Table

VIII. CONCLUSION

In this research, we used a novel approach to Twitter spam campaign detection - inter tweet similarity. We first performed crowdsourcing to detect potential spam tweets in the domains most affected by spam threats. By identifying a particular spam tweet in the "Giveaway" domain, we then moved on to use the approach the intertweet similarity to further find users involved in this particular spam campaign. We found the jaccard similarity of other tweets in the aforementioned domain with the spam tweet. If the distance was found to be less than the threshold value(0.4), then the corresponding user became a part of the spam network. As a result, we were able to create a network of spam users, without using machine learning algorithms.

Additionally, we performed Network Motif Analysis of the detected spam network in order to find the way in which the spam campaign was spread. We used FANMOD tool to do the same. Using the analysis approach, it was detected that there was a single central user who initiated the spam campaign. The spam was further spread by multiple other support users, who posted similar tweets on their accounts. The analysis result was represented in a tabular form as shown above.

IX. FUTURE SCOPE

In the future, this method can be used to detect different types of network motifs spread across Twitter.

It can also be combined with other machine learning algorithms to improvise the F-score and hence optimize the results obtained through the respective techniques.

Apart from Twitter, we can modify and optimize this technique to detect and analyze spam campaigns spread across different social networking platforms such as Instagram, Facebook etc.

REFERENCES

[1] Becchetti, L.; Boldi, P.; Castillo, C.; and Gionis, A. 2008. Efficient semi- streaming algorithms for local triangle counting in massive graphs. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 08, 1624. New York, NY, USA: ACM.

[2] Benevenuto, F. Rodrigues, T. Almeida, V. Ameida, J. and Goncalves, M. 2009. Detecting spammers and content promoters in online video social networks.In Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 09, 620 627. New York, NY, USA: ACM.

[3] Milo, R.; Itzkovitz, S.; Kashtan, N.; Levitt, R.; Shen-Orr, S.; Ayzenshtat, I.; Sheer, M.; and Alon, U. 2004. Superfamilies of Evolved and Designed Networks. Science 303(5663):15381542.

[4] D. Metzler, S. Dumais, and C. Meek, Similarity Measures for Short Segments of Text, Proceedings of the 29th European Conference on IR Research (ECIR 2007), 2007, pp. 16-27.

[5] K. Lee, J. Caverlee, S. Webb, Uncovering Social Spammers: Social Honeypots +Machine Learning, in: Proc. 33rd Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retr., Geneva, Switzerland, 2010.

[6] R. Milo, S. S. Shen-Orr, S. Itzkovitz, et al. Network motifs: Simple building blocks of complex networks. Science, 298(5594):824827, 2002.

[7] S. Wernicke and F. Rasche. FANMOD: a tool for fast network motif detection. Bioinformatics, 22(9):11521153, 2006.

[8] Chao Chena, Sheng Wena, Jun Zhanga, Yang Xiang, Jonathan Oliver, Abdul hameed Alelaiwi, Mohammad Mehedi Hassan, Investigating the deceptive information in Twitter spam, 2017.

[9] Ashish Sureka, Mining User Comment Activity for Detecting Forum Spammers in YouTube, 2011

[10] Srishti Gupta, Abhinav Khattar, Arpit Gogia, Collective Classification of Spam Campaigners on Twitter: A Hierarchical Meta-Path Based Approach, 2018. 32

[11] C. Grier, K. Thomas, V. Paxson, M. Zhang, @spam: the underground on 140 characters or less, in: Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS10, ACM, New York, NY, USA, 2010, pp. 2737.

[12] A.H. Wang, Dont follow me: Spam detection in twitter, in: Proceedings of the 2010 International Conference on Security and Cryptography, SECRYPT, 2010, pp. 110.

[13] C. Yang, R. Harkreader, J. Zhang, S. Shin, G. Gu, Analyzing spammers social networks for fun and profit: a case study of cyber criminal ecosystem on twitter, in: Proceedings of the 21st International Conference on World Wide Web, WWW12, ACM, New York, NY, USA, 2012, pp. 7180.

[14] S. Yardi, D. Romero, G. Schoenebeck, D. Boyd, Detecting spam in a twitter network, First Monday 15 (14)(2010).

[15] H. Kwak, C. Lee, H. Park, S. Moon, What is twitter, a social network or a news media? in: Proceedings of the 19th International Conference on World Wide Web, WWW10, ACM, New York, NY, USA, 2010, pp. 591600.

[16] X. Jin, C.X. Lin, J. Luo, J. Han, Socialspamguard: A data mining-based spam detection system for social media net- works, PVLDB 4 (12) (2011) 14581461.

[17] S. Ghosh, B. Viswanath, F. Kooti, N.K. Sharma, G. Korlam, F. Ben- evenuto, N. Ganguly, K.P. Gummadi, Understanding and combating link farming in the twitter social network, in: Proceedings of the 21st International Conference on World Wide Web, WWW12, ACM, New York, NY, USA, 2012, pp. 6170.

[18] H. Costa, F. Benevenuto, L.H.C. Merschmann, Detecting tip spam in location based social networks, in: Proceedings of the 28th Annual ACM Symposium on Applied Computing, SAC13, ACM, New York, NY, USA, 2013, pp. 724729.

[19] C. Chen, J. Zhang, X. Chen, Y. Xiang, W. Zhou, 6 million spam tweets: A large ground truth for timely twitter spam detection, in: IEEE ICC 2015 - Communication and Information Systems Security Symposium, ICC15 (11) CISS, London, United Kingdom, June 2015, pp. 86898694.

[20] C. Chen, J. Zhang, Y. Xiang, W. Zhou, Asymmetric Self-Learning for tackling twitter spam drift, in: The Third Interna- tional Workshop on Security and Privacy in Big Data, BigSecurity 2015, Hong Kong, Hong Kong, Apr. 2015, pp. 237242.

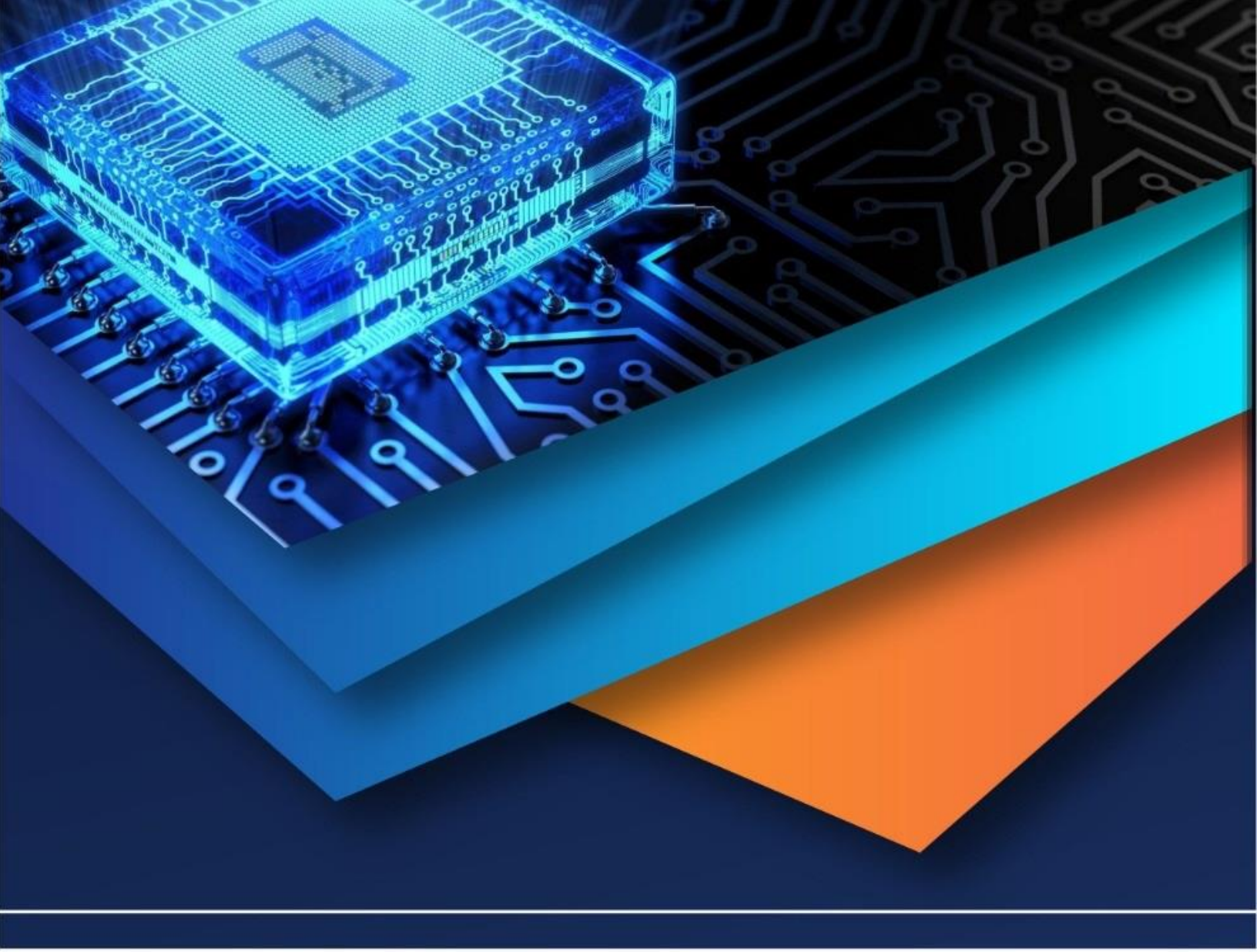
[21] E. Tan, L. Guo, X. Zhang, Y. Zhao, Unik: Unsupervised social network spam detection, in: Proceedings of 22nd ACM International Conference on Information and Knowledge Management, San Fransisco, USA, October 2013.

[22] M. Egele, G. Stringhini, C. Kruegel, G. Vigna, Compa: Detecting compromised accounts on social networks, in: Annual Network and Distributed System Security Symposium, 2013.

[23] R. Milo, S. Itzkovitz, N. Kashtan, et al. Superfamilies of designed and evolved networks. Science, 303(5663):15381542, 2004.

AUTHOR BIOGRAPHY

Arush Agarwal



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY