A Phone Numbers Based Approach Towards Social Spam Campaigns' Extermination.

Yash Mittra[1][0000−0001−6163−6174], Nimish Verma[1][0000−0002−2337−3569], Rahul Anand[2][0000−0002−1293−8421], and Bhaskar Kapoor[1]

[1] Maharaja Agrasen Institute of Technology,Delhi
[2] Bhagwan Parshuram Institute of Technology, Delhi

Abstract. The internet has changed our lives in ways unprecedented, social media being the biggest one of them. However, with the facilities social media provides, it is also a dream come true for spammers with malicious campaigns. People have misused these online social platforms to disseminate spam across the borders limiting their countries. In this research, we aim at identifying the spam campaigns (focusing on phone numbers) and their intensity on social media.

The scope of this project is limited to three of the prevailing online social media platforms: Twitter, Tumblr, and Flickr. These platforms make up for a huge part of the spam being generated by spammers online.

We start by collecting potentially spam posts from the three platforms. We collected a total of 18 million posts over a period of 4 months. We use regular expressions to prune the posts containing phone numbers. The country codes are used in detecting the potential location of origin of the post. Latent Dirichlet Allocation (LDA) is used to perform topic modeling, which results in 11 of the most prominent categories in our procured dataset. LDA is a Bayesian model built based on the probabilistic graphical model (PGM) formalization and can be flexibly integrated into other Bayesian models [1]. Keywords related to each of these categories are identified. The posts are vectorized using Bag-of-words and Tf-idf approaches and Cosine Similarity is used as a measure for classification. An in-detail analysis is carried out on the classified data. A web portal is built on which a phone number can be input and the location of the spam campaign can be identified.

Keywords: Spam Campaigns · Social Media Spams · Online Social Networks

1 Introduction

Online Social Networks (OSNs) such as Twitter, Tumblr and Flickr provide a collaboration and communication platform to millions of people on the internet[8]. In fact, in the first quarter of 2018, Twitter had almost 336 million active users[11]. The users believe that these platforms provide a more secure environment,to communicate with their friends and families, than the rest of internet. There has been evidence that these "secured" communities can become a medium to fool more people into spam campaigns[2,3]. Using fake accounts, attackers can make the trust environment of OSNs dangerous for its users, by luring users using the spam posts. We found that most of the content was generated using twittbot.net[7]. In this paper, we represent a first of its kind study on the spread of these spam campaigns over three OSNs, namely Twitter, Tumblr and Flickr. Our work is based on a large dataset of Tweets from Twitter, Blog Posts from Tumblr, and Comments on Flickr (collectively called as posts in our paper). These are permanent unless removed explicitly by the author. Through the APIs provided by these OSNs, we collected a large amount of data from each site using around 400 spam keywords. ¡posts per users per source¿ was collected

Our study of spam campaigns contains two phases. First, we analyze these posts, to filter out the ones relevant to our study. We filter tweets using popular hashtags, as an attempt to improve our labelled collection [10]. Since we are interested in the spread of these spam campaigns, we look out for presence of phone number(s) in these posts. This was done using various Regular Expressions to filter out the various formats of phone numbers followed globally. From these filtered tweets, we located the region of the corresponding spam campaigns using the country code in the phone numbers present.

In the second phase, we analyze the characteristics of the spam post we have identified. Our results provide interesting insights on the spread of spam campaigns in OSNs. We find that the category ”Contact Us” is the most popular spam campaign on Twitter and Tumblr. We also find that the most spread campaigns have around 3000 accounts posting for them. Finally, we study the cross-platform presence of the campaigns and the corresponding use of ”hashtags”.

In summary, we present in this project the first attempt to locate these spam campaigns globally and study the spread on various OSNs. We employ multiple techniques to classify the spam posts into various categories, and to study the spread of a campaign. Our subsequent analysis provides insights into the details of the spam posts, such as the hashtags used, and the number of accounts posting for a single campaign.

2 Methodology

2.1 Data Collection

Three platforms were targeted: Twitter, Tumblr and Flickr. Data was collected from each of these using APIs and Python-based-wrappers. For the data collection, three scripts were written for each of these platforms and were run periodically (once every two hours). This periodic triggering of the scripts was performed using CronTabs on a UNIX based system. Since our research targets phone numbers based spam posts, we chose a set of keywords which would be very likely to be included in spam posts containing phone numbers. These keywords were used as an input to the APIs in order to perform the data collection. We used a total of 482 keywords. Some of them are Call me, contact,

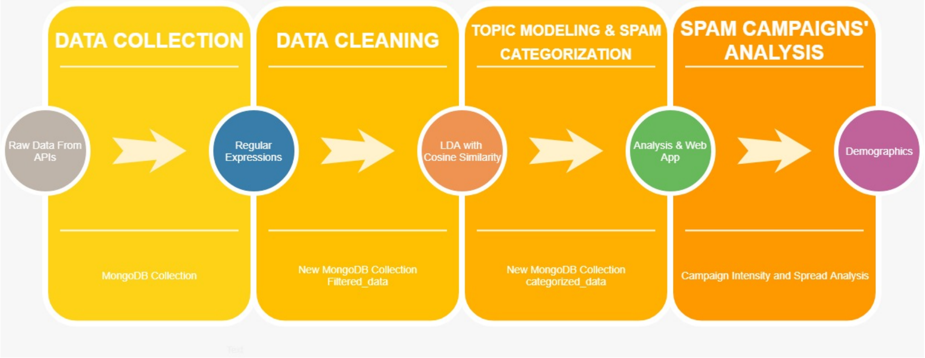

DATA COLLECTION
DATA CLEANING
TOPIC MODELING & SPAM CATEGORIZATION
SPAM CAMPAIGNS' ANALYSIS
Raw Data From APIs
Regular Expressions
LDA with Cosine Similarity
Analysis & Web App
Demographics
MongoDB Collection
New MongoDB Collection Filtered_data
New MongoDB Collection categorized_data
Campaign Intensity and Spread Analysis

Fig. 1. An overview of the methodology followed during the research.

Whatsapp, and Hit me up. The complete list of keywords that was used can be found in the project root folder on GitHub under the name keywords.txt. We used MongoDB for data storage for the reason that the data was coming from a variety of platforms and the structure of the data was tentative. Moreover, using MongoDB also helps us avoid duplicacy which might take place if the same post appears more than once from the repetitive API calls. The initial structure of the pipeline contained the following attributes which were collected from each of the APIs to maintain a consistent schema. Each of these attributes was made into a JSON object which would then be inserted into the database if the post was not a duplicate. The description of the attributes that are collected are as follows:

RT: *True or false depending upon whether the tweet is a retweet or not. For other APIs, this may be something such as "shared".*

keyword: *the query (from keywords.txt that was used to retrieve the post)*

source: *specifies the source API, Example: twitter, flickr, or Tumblr. This is always hard-coded.*

content: *specifies the text in the post/comment captured*

hashtags: *Hashtags found in the posts. If found, we separated them and added them to the JSON object here. Otherwise, this may be empty. The hashtags are present as a list of strings.*

_id: *id of the post on that specific platform*

user_id: *id of the user on that specific platform*

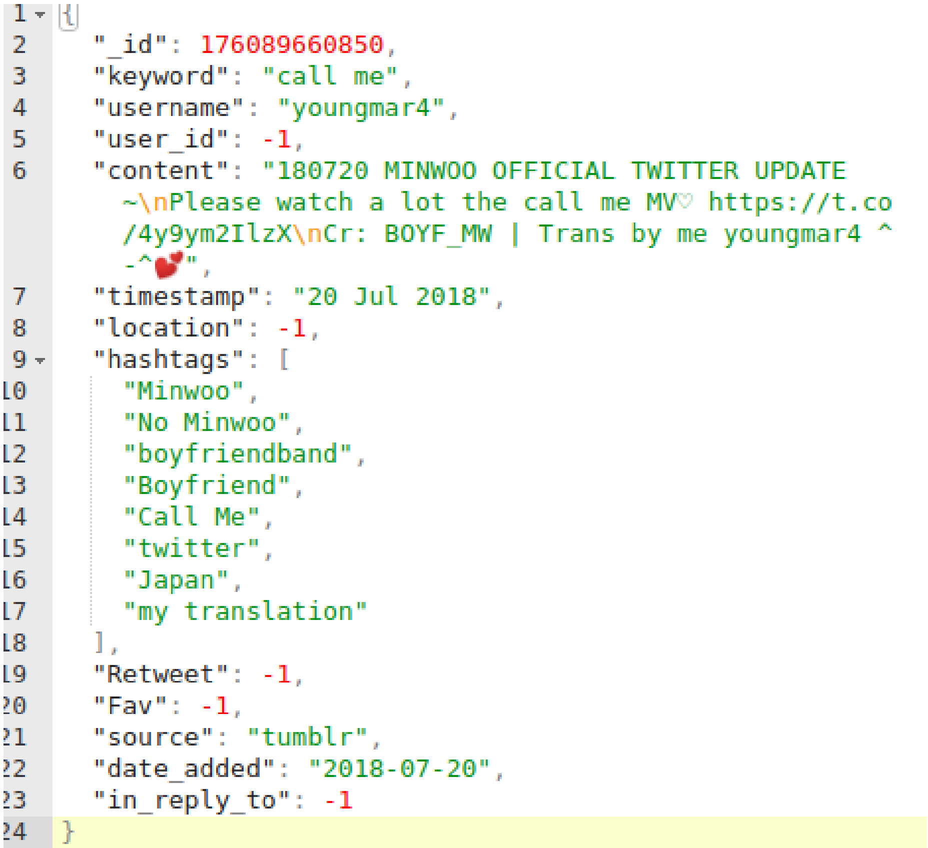
time_stamp: *time of creation of post. Format: strftime('%d %b %Y')*

in_reply_to: *(hard coded this to -1 if some APIs do not support this); useful for identifying campaigns*

fav: *favorite/like count*

date_added: *date we collected the tweet into our DB. Format: strftime('%d %b %Y') format.*

A snapshot of one of the initial json data object collected through the APIs is shown in Figure 2.



```
{
  "_id": 176089660850,
  "keyword": "call me",
  "username": "youngmar4",
  "user_id": -1,
  "content": "180720 MINWOO OFFICIAL TWITTER UPDATE
    ~\nPlease watch a lot the call me MV♡ https://t.co
    /4y9ym2IlzX\nCr: BOYF_MW | Trans by me youngmar4 ^
    -^💕",
  "timestamp": "20 Jul 2018",
  "location": -1,
  "hashtags": [
    "Minwoo",
    "No Minwoo",
    "boyfriendband",
    "Boyfriend",
    "Call Me",
    "twitter",
    "Japan",
    "my translation"
  ],
  "Retweet": -1,
  "Fav": -1,
  "source": "tumblr",
  "date_added": "2018-07-20",
  "in_reply_to": -1
}
```

Fig. 2. In total, over 19M posts were collected with following the same format.

2.2 Data Cleaning

The data is cleaned using patterns in regular expressions. Since our research focuses on phone numbers based spam, only the data containing phone numbers needed to be extracted. This data would form a new database called filtered_posts. In this database, the objects that were saved relied mostly on the schema from the main database where the tweets were mainly collected. However, we added two new attributes to this data numbers, and region. The numbers attribute was a string of comma separated phone numbers that we found in the database. At the same time, we used PyCountry to locate the location of the spam campaign using the country code for that particular region[4]. While the country code did not appear in many phone numbers found in the posts, the key areas of origination of a large number of posts were still identified. This area of origination of the post was stored against the attribute region in our database. In case that the post was unable to be located through this mechanism, the region was equated to -1. The region attribute was introduced to complement the already existing parameter of location in the json objects. The location parameter is rarely available when data is extracted through the APIs. Moreover, when it is available, most times it is inaccurate and meaningless. Hence, our best bet to finding the location of the tweets was to consider the country codes. While this may not be entirely accurate since it is possible to contribute to spam campaigns of a country even when we are present in a completely different country, it is safe to assume that the spam campaigns on Online Social Networks usually contain spam originating and contributing to the spam campaigns of the same country.

The following regular expressions were used for extracting phone numbers containing the country codes, and were our main focus.

$$r1p = r'(\backslash + \backslash d\{1,5\} - \backslash d\{3,5\} - \backslash d\{3,7\})' \tag{1}$$

$$r2p = r' \backslash + \backslash (?[\backslash d]\{3\}\backslash)?[- \backslash s]?[\backslash d]\{3\}[- \backslash s]?[\backslash d]\{3,6\}' \tag{2}$$

$$r3p = r'(\backslash + \backslash(\backslash d\{3,5\}\backslash)[- \backslash s \backslash d] \backslash d\{2,5\}[- \backslash s \backslash d] \backslash d\{2,6\})' \tag{3}$$

$$rp = r'(\backslash + \backslash d[\backslash d \backslash s-]\{5,13\} \backslash d\{2,4\})' \tag{4}$$

The regular expressions can be seen as four distinct expressions, each of which has a corresponding regular expression which is the same itself but is tailored towards accepting phone numbers with country codes. These expressions which are well-suited to detect the country code in the phone numbers are given a higher preference over their counterparts which are not tailored to process the country codes. Hence, the same preference order is maintained while matching the phone numbers against the regular expressions. These 19M posts from the original database were processed using a pymongo cursor in batches of 50K in each iteration until the cursor ran out of objects in the database.

2.3 Spam Categorization

A spam campaign is defined as a collection of multiple accounts controlled and manipulated by a spammer to spread spam on Twitter for a specific purpose[9] The cleaned data consisted of around 450,000 posts, which consisted of spams of various categories. First, the data was preprocessed wherein we removed the stop words, stemmed, and lemmatized the data. The preprocessed collection of data was then converted into a dictionary containing all the words in our database. The global dictionary is converted into a corpus using the python package gensim.

For identifying the categories, we used a hybrid approach. We used Bag of Words(BoW) and LDA to get the most prominent categories in our database[5]. The corpus previously obtained, is used here as an input to the BoW and LDA models (obtained from gensim package). The result was a list containing various sets of keywords that belonged to a single category. Thus, giving us an estimate of the topics present in our database.

Using manual identification, we made a set of top 11 categories and corresponding keywords. This structure is later used to compute the similarity of a post to each category. We obtained the following categories:

1. Selling
2. Pornography
3. Social Media Promotion
4. Contact Us
5. Banking
6. Politics
7. MOOCs
8. Appointment
9. Racism
10. Legal
11. Give Away

Finally, we clustered the posts among these categories. For this we used, cosine similarity using tf-idf weights[6]. A post was assigned to the class, for which it had the highest cosine similarity with. The end result was, a corresponding category (spam class) for each post. The clustered database was later used for analysis.

$$\cos(\mathbf{t},\mathbf{e}) = \frac{\mathbf{t}\mathbf{e}}{\|\mathbf{t}\|\|\mathbf{e}\|} = \frac{\sum_{i=1}^{n}\mathbf{t}_i\mathbf{e}_i}{\sqrt{\sum_{i=1}^{n}(\mathbf{t}_i)^2}\sqrt{\sum_{i=1}^{n}(\mathbf{e}_i)^2}} \tag{5}$$

A high cosine value indicates that a post is closely related to the topic model and thus a good candidate for being included in that category.

3 Results

3.1 Number of posts per category with percentage

The first analysis was done to calculate number of posts per category with percentage. The analysis helps us understand what category of posts contribute to least and most amount of spam on social media. To calculate number of posts per category, we simply iterated on all the posts and kept on incrementing the count by one if the post lie in an existing category. We obtain a list of tuples containing category and number of posts in that category. We then sort it in descending order on the number of posts. The highest category percentage was Contact Us followed by Selling

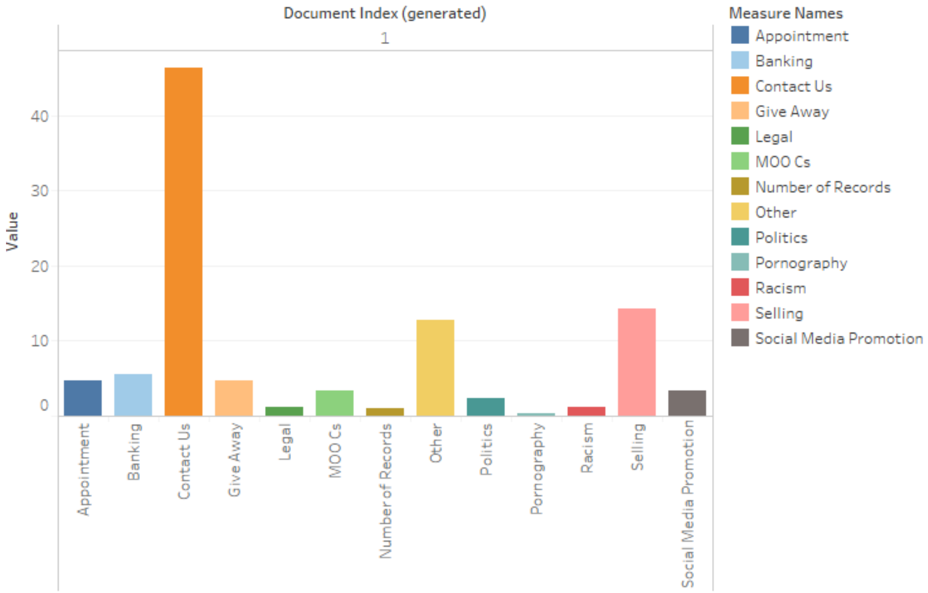
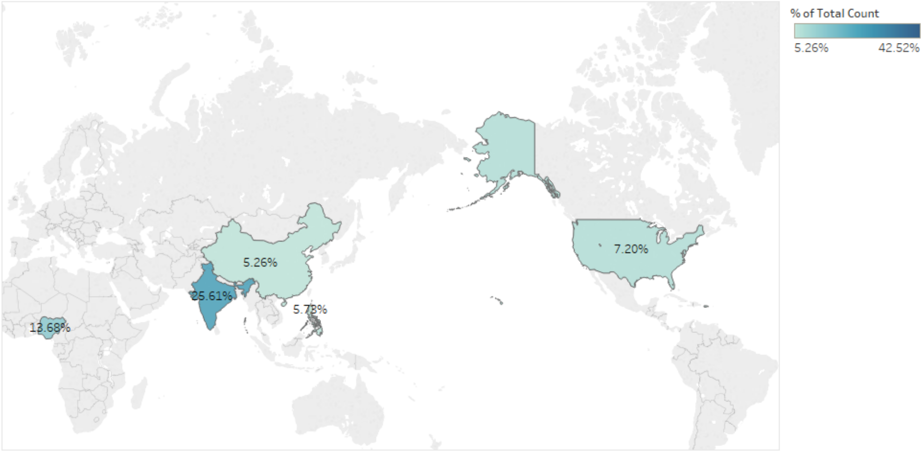


Fig. 3. Percentage of post per category

3.2 Countries generating most spam

The second analysis was done to identify the percentage of spam posts by country. This analysis helps us identify the countries from where most and least spam originates. To calculate this, we iterated on all the posts and increment the count by one if the post comes from an existing region. We obtain a list of tuples containing region and the number of posts from that region. We then sort it in descending order on the number of posts. The country generating most spam was India followed by Nigeria and United States



Map based on Longitude (generated) and Latitude (generated). Color shows % of Total Count. Details are shown for Country.

Fig. 4. Map hightlighting the most spam-generating countries

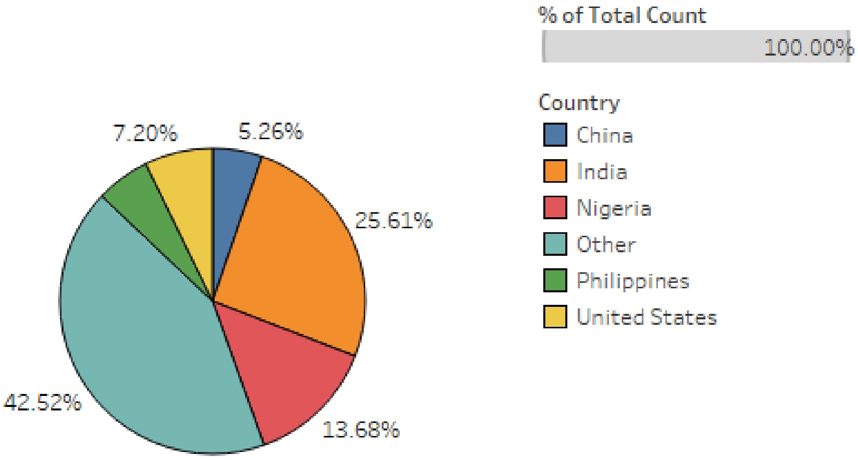


Fig. 5. Pie Chart depicting percentage of spam from countries

3.3 Top 10 Spammers along with their Spam Categories and Number of Spam Posts

The third analysis we carried out was to recognise top spam accounts along with their category of spam and number of posts. Again, we iterated on all the posts and kept incrementing the posts count if the user already existed in our dictionary. We obtain a list of tuples containing user id, a dictionary of subsequent category of spam and number of posts posted by them. We then sort the list in descending order to identify top ten spammers in world. Top spammer was user ID 3325130122 who spammed 2154 posts under categories Legal, Contact Us and Other.

Table 1. Top spamming users.

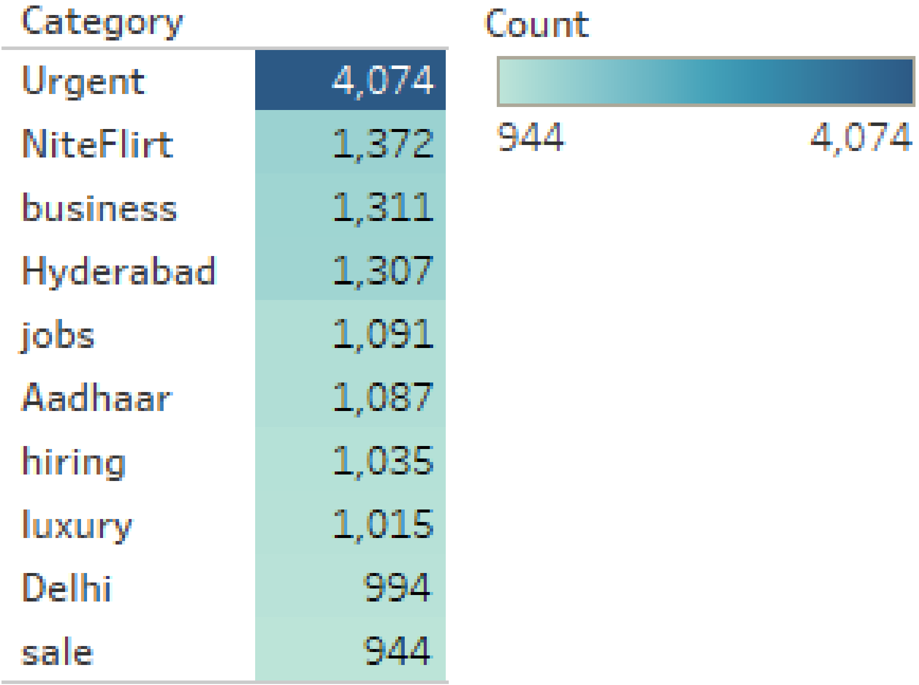
USER ID	CATEGORIES	COUNT
3325130122	Contact Us, Legal, Other	2154
78661839	Appointment, Banking, Contact Us, Give Away(...)	1379
46528834@N05	Appointment, Banking, Contact Us, Other(...)	1035
4148799826	Appointment, Contact Us, Give Away, Other...	646
2208888664	Contact Us, Selling	521
17872077	Appointment, Banking, Contact Us, Give Away(...)	494
145931959	Appointment, Banking, Contact Us, Give Away(...)	414
877585625876221953	Appointment, Selling	374
3297760074	Give Away, Legal, Racism	351
85741735	Appointment, Banking, Contact Us, Give Away(...)	317

3.4 Total number of users that generated this much spam

The fourth analysis is carried out to calculate the numbers of users that generated spam. We simply find the length of list obtained in previous analysis to get the total number of users who generated this spam. The total number of users were 262682

3.5 Top hashtags in the dataset

The fifth analysis is carried out to calculate the top hashtags appearing in these posts. To calculate this, we iterate the posts and increment the count if hashtag already exists in dictionary. We obtain a list of tuples containing hashtags and number of times they occured in all posts. The list is sorted in descending order on the number of occurrences to identify the top hashtags. The top three hashtags were Blood, need, urgent with 3402, 2716 and 2581 mentions respectively.



Category	
Urgent	4,074
NiteFlirt	1,372
business	1,311
Hyderabad	1,307
jobs	1,091
Aadhaar	1,087
hiring	1,035
luxury	1,015
Delhi	994
sale	944

Count
944
4,074

Fig. 6. Top Hashtags

3.6 Most Prominent Spam Category as per Platform

The sixth analysis is carried out the calculate the most prominent spam category platform. In this analysis we aim to identify which platform. We again iterate the posts to and create a dictionary of social platforms. For each social platform, we have category wise spams inside another dictionary which are incremented if the category exists. After that, we divide count of posts in each category by the total number of posts on that platform to find out the percentage of spam that category constitutes on that particular platform. While the highest spammed category in Flickr and tumblr were Other, Give away, and selling, twitter had 'Social Media Promotion' and Politics as most spammed categories.

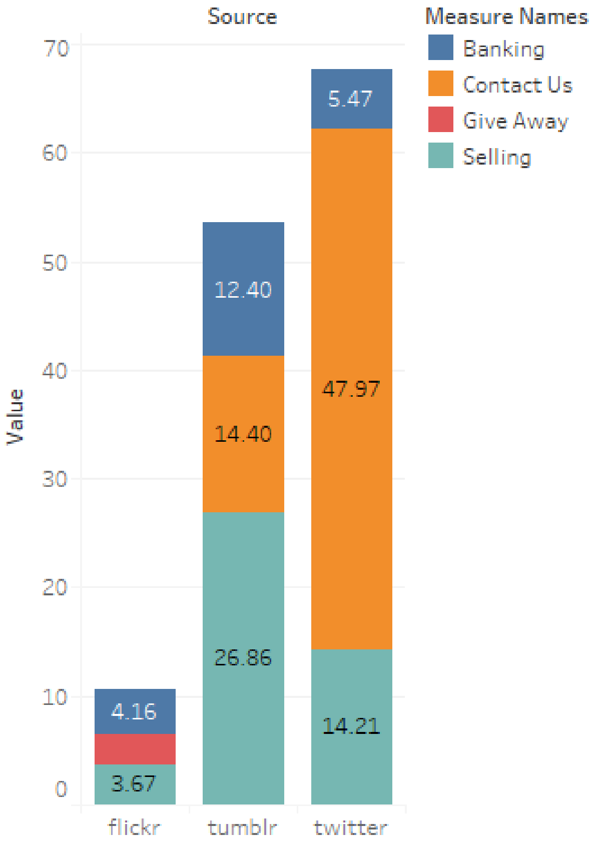


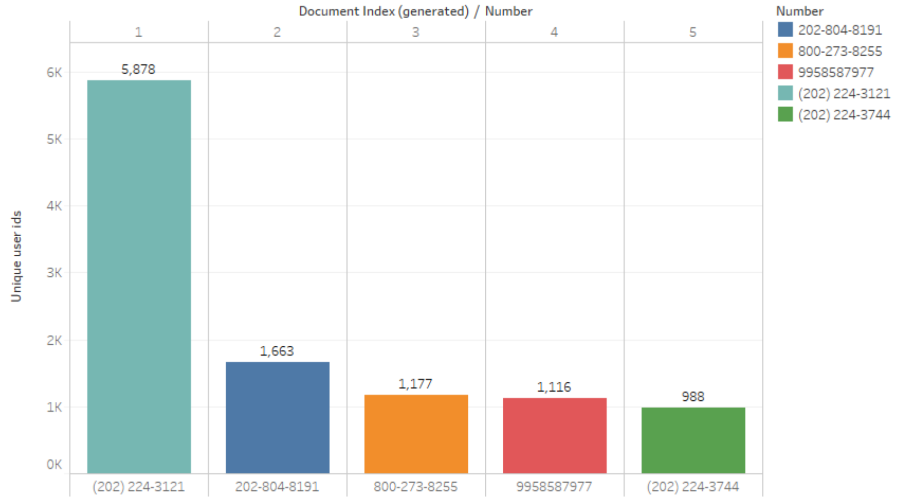
Fig. 7. Most Prominent Spam Category

3.7 Same phone number in posts from multiple platforms

The seventh analysis is carried out to identify the total number of phone numbers common across multiple platforms. We again iterate the posts and create a new set for each platform. Each set contains phone numbers found in posts on respective platforms. We then calculate the number of phone numbers common in different sets. Total number of phone numbers that were common across different platforms were 526

3.8 Maximum number of people related to same campaign

The eighth analysis is carried out to identify the total number of users related to same campaign. To calculate this, we iterate the posts and check if a phone number, user id pair exists in our degree. We increment the count by one if a different user id is found for same number. We get a list of tuples of phone number and count of users as final output. This list is then sorted on count of user id. The phone numbers '202-224-3121' and '(202) 224-3121' topped the list with mentions from 2970 and 2908 accounts respectively.



Sum of Unique user ids for each Number broken down by Document Index (generated). Color shows details about Number.

Fig. 8. Top Spammers

4 Conclusion

Millions of people are exploited by spammers all over the world. This calls for a need to successfully identify the spam from the non-spam. If we are able to detect spam posts while limiting the false-positives, we can in fact hinder these spammers from exploiting the social networking platforms. This can help in various ways. Lesser people will fall prey to these spammers who have decided to exploit these innocent people for their personal or financial gain.

Our work provides a detailed analysis of the spam generated online and it clearly shows the contribution of some of the major countries towards spam promulgation. Special filters can be applied on posts that originate in these areas in order to protect social platforms from malicious posts.

We also built a web application where the users can input the phone number and check the location of origination of the spam post. In cases where the location shows up, it means that the post has been classified as spam and processed through our pipeline. This is in prospect a sign towards eliminating the spam campaigns from social media completely, and to protect the interests of innocent users of the web.

5 Future Work

The current analysis and web application has been realized in a manual approach where we had to use manual human expertise for some of the topic modeling part and feeding the data into the web application database. Our future work on this research involves automating the topic modeling completely using an algorithm such as the tri-gram algorithm. For complete automation, a pipeline can be built which would feed the data directly into the web application's database on an hourly basis. This would further help in maintaining an up-to-date database of spam posts which the legitimate users can exploit in order to save themselves from the decoys set up by malicious users.

We may also change the JSON object structure to include the language attribute which can help us in even more accurately determining the origin of the malicious posts.

References

1. Wei W, Joseph K, Liu H, Carley KM (2016) Exploring characteristics of suspended users and network stability on twitter. Soc Netw Anal Min 6:51
2. Users of social networking websites face malware and phishing attacks. Symantec.com Blog
3. Zeus botnet targets facebook. http://blog.appriver.com/2009/10/zeus-botnet-targets-facebook.html
4. PyCountry Library. https://pypi.org/project/pycountry/
5. https://towardsdatascience.com/combing-lda-and-word-embeddings-for-topic-modeling-fe4a1315a5b4
6. Document Similarity in Information Retrieval https://courses.cs.washington.edu/courses/cse573/12sp/lectures/17-ir.pdf
7. Srishti Gupta, Dhruv Kuchhal, Payas Gupta, Mustaque Ahamad, Manish Gupta, and Ponnurangam Kumaraguru. 2018. Under the Shadow of Sunshine: Characterizing Spam Campaigns Abusing Phone Numbers Across Online Social Networks. In Proceedings of the 10th ACM Conference on Web Science (WebSci '18).

8. Hongyu Gao, Jun Hu, Christo Wilson, Zhichun Li, Yan Chen, and Ben Y. Zhao. 2010. Detecting and characterizing social spam campaigns. In Proceedings of the 10th ACM SIGCOMM conference on Internet measurement (IMC '10).
9. Zi Chu, Indra Widjaja, and HainingWang. 2012. Detecting social spam campaigns on twitter. In International Conference on Applied Cryptography and Network
10. Fabricio Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. 2010. Detecting spammers on twitter. In Collaboration, electronic messaging, anti-abuse and spam conference (CEAS), Vol. 6. 12.
11. Twitter MAU in the United States 2018 — Statistic, https://www.statista.com/statistics/274564/monthly-active-twitter-users-in-the-united-states/. Last accessed 4 Oct 2018