Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.817 – 824

RESEARCH ARTICLE

Mail_Alert: Online Suspicious URL Detection of Tweets from Twitter Public Timeline

SPOORTHI K
M.Tech Computer Network &Engineering
Reva Institute of Technology, Kattigenahalli, Bangalore 560064, Karnataka, India
Email: gowdaspoorthi99@gmail.com

SARVAMANGALA D R
Senior Assistant Professor, Dept. of Computer science & Engineering
Reva Institute of Technology, Kattigenahalli, Bangalore 560064, Karnataka, India
Email: sarvamangala.dr@revainstitution.org

***Abstract:** Twitter, a famous social networking site where thousands of users use it to tweet to the world, is prone to spam, phishing, and malware distribution. Tweets are the atomic building blocks of Twitter, 140-character status updates with additional associated metadata. People tweet for a variety of reasons about a multitude of topics. Traditional spam detection scheme for twitter are ineffective against feature fabrications or consume much time and resources. Conventional suspicious URL detection schemes utilize several features including lexical features of URLs, URL redirection, HTML content, and dynamic behavior. However, evading techniques such as time-based evasion and crawler evasion exist. In this paper, we propose a suspicious URL detection system for Twitter in which numerous tweets from the Twitter public timeline is collected and dynamic trained classifier is been built to classify among suspicious and the real ones. Timelines are collections of Tweets, ordered with the most recent first. Evaluation results show that our classifier accurately and efficiently detects suspicious URLs. A near real-time system for classifying suspicious URLs in the Twitter stream. In this paper I propose to block the malicious URLs and provide mail alert for malicious URLs occur in the twitter stream.*

Keywords—Twitter, Trained SVM classifier, share URLs, mail_alert

I. Introduction

Twitter is an online social networking and micro blogging service that enables users to send and read short 140-character text messages, called *"tweets"*. Registered users can read and post tweets, but unregistered users can only read them. Users access Twitter through the website interface, SMS, or mobile device app. Twitter now boasts 100 million active users, half of which tweet to the site on a daily basis. There is an 82 percent increase since the beginning of the year, Twitter representatives said in an email. Studies have shown that Twitter positively contributes to a site's social popularity, so it's no surprise that most of the world's top sites feature a Twitter button. Microsoft and Twitter have recently renewed their search agreement. Static or dynamic crawlers are used and may be executed in virtual machine honeypots, such as Capture-HPC, HoneyMonkey, and Wepawet, to investigate newly observed URLs. Twitter is a very simple service that is rapidly becoming one of the most talked-about social networking service providers. When you have a Twitter account, you can use the service to post and receive messages to a network of contacts, as opposed to send bulk email messages. You can build your network of contacts, and invite others to receive your Tweets, and can follow other members' posts. Twitter makes it easy to opt into or out of networks. Additionally, you can choose to stop following a specific person's feed. The web crawler does not accept a list of Seed URLs. The efficiency of the Web crawler can be improved by making it a multithreaded web crawler. Analyzing requires some manual work.

For instance, because static crawlers usually cannot handle JavaScript. Or Flash, malicious servers can use them to deliver malicious content only to normal browsers. Malicious servers can also employ temporal behaviors—providing different content at different times to evade an investigation. Online social networking service is a platform to build social networks or social relations among people who, for example, share interests, activities, backgrounds or real-life connections. A social network service consists of a representation of each user (often a profile), his social links, and a variety of additional services. Social networking is web-based services that allow individuals to create a public profile, to create a list of users with whom to share connection, and view and cross the connections within the system. Most social network services are web-based and provide means for users to interact over the Internet, such as e-mail and instant messaging. Online social network sites are varied and they incorporate new information and communication tools such as, mobile connectivity, photo/video/sharing and blogging. Online community services are sometimes considered as a social network service, though in a broader sense, social network service usually means an individual-centered service whereas online community services are group-centered. Social networking sites allow users to share ideas, pictures, posts, activities, events, and interests with people in their network.

II. Proposed System

The SVMLight tool is machine learning, support vector machines .SVMs, also called as support vector networks are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. It has scalable memory requirements and can handle problems with many thousands of support vectors efficiently. The software also provides methods for assessing the generalization performance efficiently.

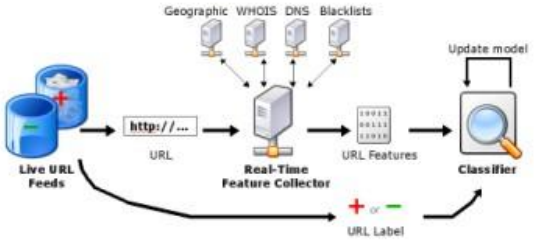

Fig 1 Depicts machine learning techniques to detect malicious URLs

Description of Data (SVM-light)

Uncompressing the archive *url_svmlight.tar.gz* will yield a directory url_svmlight/ containing the following files:

- FeatureTypes - A text file list of feature indices that correspond to real-valued features.
- DayX.svm (where *X* is an integer from 0 to 120) -- The data for day *X* in SVM-light format. A label of +1 corresponds to a malicious URL and -1 corresponds to a benign URL.

A feature vector is an n-dimensional vector of numerical features that represent some object. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (e.g. the same measurement in both feet and meters) then the input data will be transformed into a reduced representation set of features. To label the training vectors, we use the Twitter account status; URLs from suspended accounts are considered malicious whereas URLs from active accounts are considered benign. We periodically update our classifier using labeled training vectors.

In this section discusses issues related to algorithm. Three steps used in our offline supervised learning algorithm

- Case-A: Frequent URL with similar domain names and from same IP address.
- Case-B: Reoccurrences of redirect chains in URLs (entry points)
- Case-C: Check whether same URL is Posted to other users (followers) from same IP.

III. System Design

Our system consists of six components: data collection, feature extraction, training, classification, detecting suspicious URLs, and Mail_Alert.

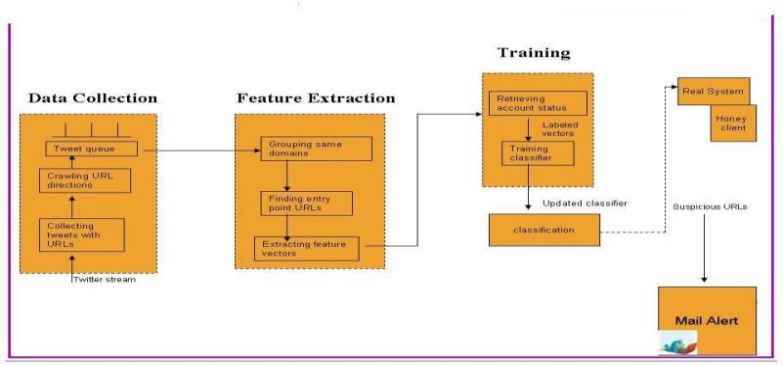

Fig. 2 System Architecture

A. Data collection

The Streaming API is the real-time sample of the Twitter. This API is for those developers with data intensive needs. If you're looking to build a data mining product or are interested in analytics research, the Streaming API is most suited for such things. This requires establishment of a long-lived HTTP connection and maintain that connection. The streaming process gets the input Tweets and performs any parsing, filtering, and/or aggregation needed before storing the result to a data store. The HTTP handling process queries the data store for results in response to user requests.

Get/statuses/mentions_timeline --Returns the 20 most recent mentions (tweets containing a users' @screen_name) for the authenticating user. The timeline returned is the equivalent of the one seen when you view your mentions on twitter.com. This method can only return up to 800 tweets.

B. Feature Extraction

Our dataset contains the following features extracted from each of the profiles the tweets, time of publication, language, geo position and Twitter client. The first feature, the tweet, is the text published by the user, which gives us the possibility of determine a writing style, very characteristic of each individual. The time of publication helps determining the moments of the day in which the users interact in the social network. The language and geo position also help filtering and determining the authorship because users have certain behaviors which can be extrapolated analyzing these features.

Background-In order to build a suspicious URL detector, first we need to equip ourselves with the right tools and methods. Machine learning is one such tool where people have developed various methods to classify. Classifiers may or may not need training data but machine learning classifiers, Support Vector Machine need it. Classifiers require training data and hence these methods fall under the category of supervised classification.

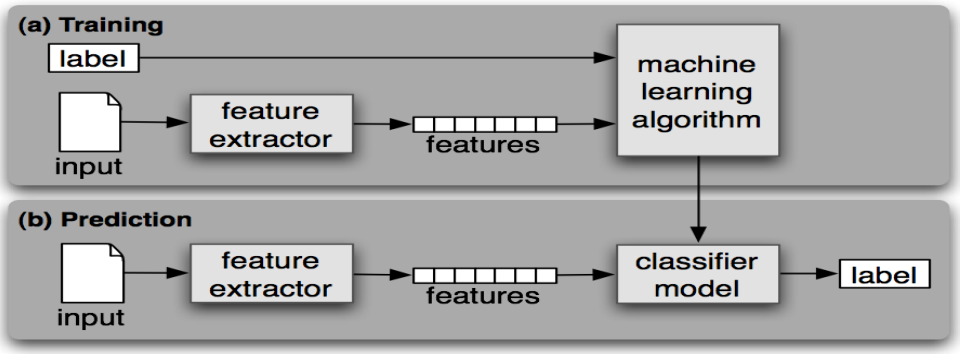

Fig.3 Supervised Classification

Training the Classifiers-The classifier need to be trained and to do that, we need to know 3 information

a) Frequent URL with similar domain names and from same IP address.
b) Reoccurrences of redirect chains in URLs (entry points)
c) Check whether same URL is Posted to other users(followers) from same IP .

Preprocess tweets

a) *Lower Case* - Convert the tweets to lower case.
b) *URLs* - I intend to follow the shortened urls and determine the content of the site, so we can stream all of these URLs via regular expression matching or replace with generic word URL.
c) *@username* - we can eliminate "@username" via regex matching or replace it with generic word AT_USER.
d) *#hashtag* - hash tags can give us some useful information, so it is useful to replace them with the exact same word without the hash. E.g. *#nike* replaced with '*nike*'.
e) *Punctuations and additional white spaces* - remove punctuation at the start and ending of the tweets. E.g: ' *the day is beautiful!* ' re

Feature Vector Feature vector is the most important concept in implementing a classifier. A good feature vector directly determines how successful your classifier will be. The feature vector is used to build a model which the classifier learns from the training data and further can be used to classify previously unseen data. Gender identification example --Male and Female names have some distinctive characteristics. Names ending in *a*, *e* and *i* are likely to be female, while names ending in *k*, *o*, *r*, *s* and *t* are likely to be male. So, you can build a

classifier based on this model using the ending letter of the names as a feature. Similarly, in tweets, we can use the presence/absence of words that appear in tweet as features. In the training data, consisting of positive, negative and neutral tweets, we can split each tweet into words and add each word to the feature vector. Some of the words might not have any say in indicating the sentiment of a tweet and hence we can filter them out. Adding individual (single) words to the feature vector is referred to as 'unigrams' approach , similarly , two words feature vector is referred to as bigram approach.

As we process, each of the tweets, we keep adding words to the feature vector and ignoring other words. Let us look at the feature words extracted for the tweets.

Example 1: " @Sre2014 I would like to work with http://www.mathworks.com/matlabcentral/answers/89361-how-to-create-feature-vector "

if this is the tweet posted then, the feature words could be http://, mathworks, .com.

Example 2: "@Sri201 post: thanks pls visit casino-services.com to get free casino services " if this is the tweet posted, the feature words will be casino, .com

The entire feature vector will be a combination of each of these feature words. For each tweet, if a feature word is present, we mark it as 1, else marked as 0. Instead of using presence/absence of feature word, you may also use the count of it, but since tweets are just 140 chars, I use 0/1.

Given any new tweet, we need to extract the feature words as above and we get one more pattern of 0s and 1s and based on the model learned, the classifiers predict the tweet sentiment. It's highly essential for you to understand this point and I have to tried to make it as simple as possible.

Tweet variable are the possible keywords present in malicious tweets [([‘http://’, ’.com’, ‘casano’, ‘rasgas’,’bfsecurity.co.in’)]

Our big feature vector now consists of all the feature words extracted from tweets. Let us call this "featureList", now we need to write a method, which gives us the crisp feature vector for all tweets, which we can use to train the classifier.

Feature List is shown here [(http//, .co.in, .com, casano, bfsecurity, rasgas)]

SVM Support Vector Machines (SVM) is pretty much the standard classifier which is used for any general purpose classification. As the earlier methods, explaining how SVM works will itself take an entire post. Please refer to the Wikipedia article on SVM to understand how it works. I will use the libsvm library (written in C++ and has a python handle) implemented by Chih-Chung Chang and Chih-Jen Lin to instantiate SVM. Detailed documentation of the python handle can be read in the libsvm.tar.gz extracted folder.

When you build a twitter sentiment analyzer, the input to your system will be a user enters keyword. Hence, one of the building blocks of this system will be to fetch tweets based on the keyword within selected time duration.

The most important reference to achieve this is the Twitter API Documentation for Tweet Search. There are a lot of options that you can set in the API query and for the purpose of demonstrating the API, use only the simpler options. Important to note Twitter does not allow you to fetch tweets older than 7 days.

C. Training

The training component has two sub components: retrieval of account statuses and training of the classifier. Because we use an offline supervised learning algorithm, the feature vectors for training are relatively older than feature vectors for classification. To label the training vectors, we use the Twitter account status URLs from suspended

accounts are considered malicious whereas URLs from active accounts are considered benign. We periodically update our classifier using labeled training vectors.

D. Classification

The classification component executes our classifier using input feature vectors to classify suspicious URLs. When the classifier returns a number of malicious feature vectors, this component flags the corresponding URLs and their tweet information as suspicious. The classifier used is SVM Light

- It solves classification and regression problems. For multivariate and structured outputs use SVM^{struct}.
- It solves ranking problems (e. g. learning retrieval functions in *STRIVER* search engine).
- can train SVMs with cost models and example dependent costs
- allows restarts from specified vector of dual variables
- handles many thousands of support vectors
- handles several hundred-thousands of training examples

E. Detecting Suspicious URLs

In this module, we proposed a new suspicious URL detection system for Twitter. Unlike the conventional systems, proposed system is robust when protecting against conditional redirection, because it does not rely on the features of malicious landing pages that may not be reachable. Instead, it focuses on the correlations of multiple redirect chains that share the same redirection servers. We introduced new features on the basis of these correlations, implemented a near real time classification using these features.

F. Mail_Alert

In this module, we enhance our system by providing mail alert system. Though the suspicious URLs are detected in an efficient way, it is unknown to the twitter users. Thus a Mail_Alert system is generated for providing an alert before the usage of the malicious URLs.

IV. Implementation

Our goal is to develop a suspicious URL detection system for Twitter that is robust enough to protect against conditional redirections. Consider a simple example of conditional redirections, in which an attacker creates a long URL redirect chain using a public URL shortening service, such as bit.ly and t.co, as well as the attacker's own private redirection servers used to redirect visitors to a malicious landing page. The attacker then uploads a tweet including the initial URL of the redirect chain to Twitter.

Later, when a user or a crawler visits the initial URL, he or she will be redirected to an entry point of the intermediate URLs that are associated with private redirection servers. Some of these redirection servers check whether the current visitor is a normal browser or a crawler. If the current visitor seems to be a normal browser, the servers redirect the visitor to a malicious landing page. If not, they will redirect the visitor to a benign landing page. Therefore, the attacker can selectively attack normal users while deceiving investigators. Thus shows how the benign URL and the malicious URL are classified which leads to the detection of attacker and block the malicious URL and prevents system disaster.

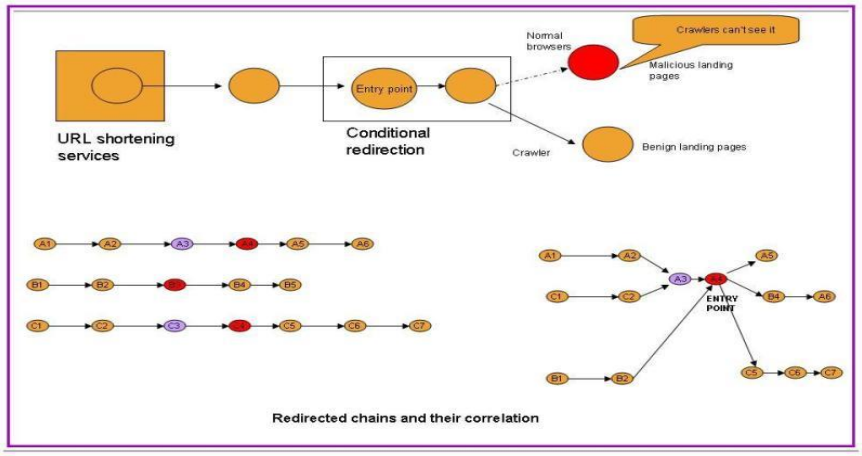

Fig 4 Rediected chains and their correlation

V. Discussion

Online detection--As per the proposed system, we present a new method of streaming the tweets from the twitter using streaming APIs provided by the twitter developers themselves and thus collect the tweets with URLs then detect for suspicious, at the same time train the classifier for detecting the malicious URL that is been tweeted by the attacker or hacker dynamically thus overcoming the disadvantages of the static or dynamic web crawlers. Training data and the features selected for use in the classifier impacts the accuracy of your classifier the most. Look up on the mentioned training data resources already available to train your classifier i.e. feature list. Thus, extract the tweets for a particular keyword. Clean the tweets and run the classifier on it to extract the labels. Then build a simple web interface which facilitates the user to login and watch the result dynamically in the browser. Crawlers do not read the image format text and text between the script tags like JavaScript or Flash Player.

VI. Performance Evaluation

Previous suspicious URL detection systems are weak at protecting against conditional redirection servers that distinguish investigators from normal browsers and redirect them to benign pages to cloak malicious landing pages its disadvantage is time consuming and less detection accuracy.

VII. Conclusion

A new method of detecting suspicious URLs by streaming the tweets from the twitter using streaming APIs provided by the twitter developers themselves and thus collect the tweets with URLs then detect for suspicious, at the same time train the classifier for detecting the malicious URL that is been tweeted by the attacker or hacker dynamically thus overcoming the disadvantages of the static or dynamic web crawlers. The login page enables you to simply login to the account and starts detecting the suspected URLs provided on the browser. The browser keeps refreshed every 5secs dynamically. The suspicious URLs could be listed in a separate list. This classification is done by the trained classifier. The beauty of the classifier is that we can add and even delete the suspicious URLs to and from the list. The login session could be recorded in the background for future references which contains the information about the user login time, detection time, errors if occurred. The same method of detection could also be applied to other social networking sites like Google plus, MySpace.

References

[1] Saranya, Udaya Kumar.V WarningBird MailAlert Based Malicious URLs Blocker System in Twitter in IJCSMC, 2014
[2] SLee and J Kim, "Warningbird: detecting suspicious urls in twitter stream," in proc. NDSS, 2012.
[3] F. Klien and M. Strohmaier, "Short links under attack: geographical analysis of spam in a URL shortened network," in Proc. ACMHT, 2012.
[4] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in Proc. WWW, 2010.
[5] Justin Ma, Alex Kulesza, Mark Dredze, Koby Crammer, Lawrence K. Saul, and Fernando Pereira, Exploiting Feature Covariance in High-Dimensional Online Learning in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 493-500, Sardinia, Italy, May 2010.
[6] https://dev.twitter.com/issues
[7] http://www.sysnet.ucsd.edu/projects/url/
[8] http://ravikiranj.net/drupal/201205/code/machine-learning/how-build-twitter-sentiment-analyzer

