Generating security intelligence through social network sentiment analysis

Ariel Rodriguez
Graduate School / Faculty of Information Science and Electrical Engineering, Kyushu University
Fukuoka, Fukuoka
roda@kyudai.jp

Koji Okamura
Graduate School / Faculty of Information Science and Electrical Engineering, Kyushu University
Fukuoka, Fukuoka
oka@ec.kyushu-u.ac.jp

ABSTRACT

Cybersecurity has moved to the forefront of the technology world in recent years with the increase in number and sophistication of attacks. Even though security has become such a crucial aspect of all organizations, the security industry still largely holds a defensive stance where reacting to attacks after they have occurred is more common than proactively finding ways to counter these threats. There are many data sources such as social networking sites, security news sites, and blogs that can be used to improve this situation and create solutions that help prepare for attacks before they occur. In this paper, we present a framework that performs sentiment analysis on security based tweets with the aim to provide relevant security information that can be used by analysts or security devices.

KEYWORDS

Text Mining, Computer Security, Machine Learning, Security Analytics, Social Networking Services

1 INTRODUCTION

Cybersecurity awareness and prevention have moved to the forefront of the technology world over the last few years with Large scale attacks and malicious activities occurring more frequently than ever before [4]. That has resulted in a defensive atmosphere in the industry which often times has to react to attacks rather than proactively finding ways to counter these threats.

To create an industry where companies are moving off the back foot and are able to more proactively defend themselves many components are required, but one of the most critical factors that will allow this to happen and which makes it possible is data. If there is not enough information available in the public domain, it can cause a reactive stance by companies making them only respond when attacked. That is not preferred since in Rowe et al. [20] they put forth that a proactive strategy incurs fewer security compromises as opposed to reactive strategies. Luckily with the rise of the internet many data resources have become available to us that can be used to help turn around the current trend of reacting to attacks once they have happened, and start creating an industry which attempts to be ready for attacks before they happen.

These data resources include Open Source Intelligence such as social networking sites, security news sites, blogs and various types of alerts, as well as close-source intelligence collected by public or private organizations. This data can be used on its own or can also be used in conjunction with other traditional types of data such as raw network traffic, Firewall logs, IPS/IDS alerts, etc. From this data we can get a better understanding of what is happening at a network traffic level. By combining all this data we can not only get a good outlook of what is happening on both a technological level but also on a socio-economic level, this is important since social unrest can often drive online and real world actions [8].

The hacktivist group Anonymous show an excellent example of this; they are a decentralized group known for its cyber attacks against governments, religious groups and companies which they feel do not morally align with their values [15]. This group has released videos on youtube as well as messages on social media and forums before attacking certain groups [3]. Even though this information may not always be legitimate, there are times when it is, and if we can efficiently mine the various open data sources that are available, we can make more informed decisions and begin to take steps to mitigate an attack earlier.

Out of all these data sources, Twitter is a particularly interesting and useful open source information tool. Twitter has a huge user base with 330 million monthly active users that post approximately 500 million tweets per day [1]. That gives us a vast amount of data to analyze. That has been done previously with studies using Twitter to look at natural disasters, terrorist attacks and political events [2, 5, 11, 21]. Twitter also has a close relationship with cyber security events. Firstly, there are many cyber security companies that post updates and information through Twitter as well as researchers who post vulnerabilities they have discovered. This information alone can be used to prepare for attacks by having the latest information available to a larger group as fast as possible. In some cases, large companies may be informed of vulnerabilities before they are reported, but often times small and/or medium sized companies are not, meaning they stand to benefit from this type of open data sharing.

Many hacktivist groups use Twitter for many different uses, they have been known to use it as an organizational method to inform followers of rallies and methods to conduct cyber attacks. In a coordinated attack on financial and Government institutions carried out by the group Anonymous called operation payback [16] tweets were posted on the @Payback_Op Twitter account which informed followers of what site to attack at a specific time and also gave a link to a DDoS tool called the "low orbit ion cannon" that could be used to attack www.visa.com. Apart from using it for attack coordination they also use it as a fear mongering system to disseminate warnings to groups they want to intimidate or plan to attack. Because of these reasons we believe that Twitter is a very good resource to acquire different kinds of cyber security information, both technological/vulnerability based and social sentiment based.

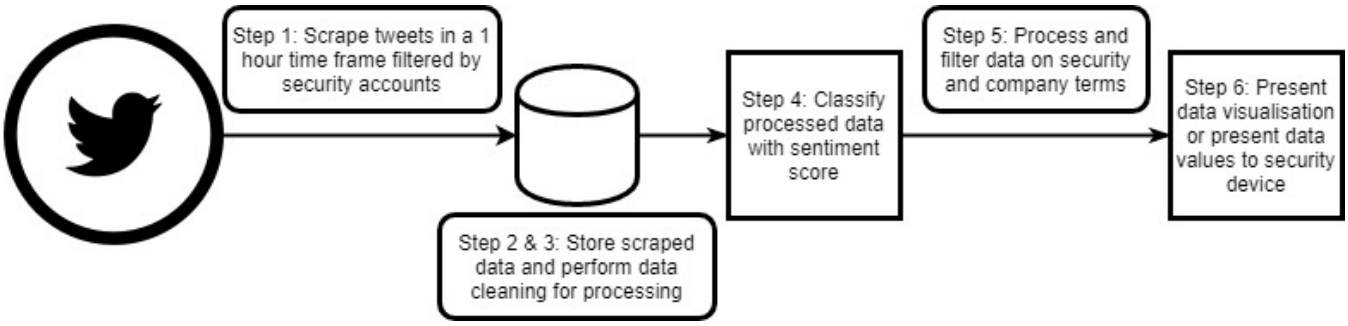

Figure 1: Data flow of system

With the number of information sources proliferating, analysts can not easily, thoroughly or efficiently comb through all sites and analyze them [22]. That is especially relevant when time is a factor and minutes can mean the difference between an attack getting through or being stopped.

In this paper, we present a framework that performs text mining and sentiment analysis on security based tweets with the aim to provide cyber security field relevant information. Text mining is the process of extracting information from text; this information can give valuable insight and understanding into the area this text is addressing. Text mining can involve more than one process such as, cleaning text, storing, finding patterns and evaluation. Sentiment analysis is a part of text mining which tries to identify the sentiment or attitude of the text. In the case of written text, it can be classified as positive, negative or neutral based on the words and frequencies of words used. The information we derive from using these techniques can be provided to analysts themselves for inspection and allows them to more easily understand the current sentiment of Twitter's security accounts and make decisions based on this information. It can also be used as an input to another application or device such as a Firewall, IDS/IPS or Antivirus which can use the information as a data point to make decisions from, and either trigger an alert, change a firewall rule or block a file.

Our framework acts as an analysis tool for cyber security tweets. It provides a general analysis of the number and type of tweets including sentiment analysis to gauge social opinion. Firstly we periodically scrape Twitter for security relevant tweets which we then process. To conduct sentiment analysis on these tweets we have chosen to use an artificial intelligence approach and train a machine learning model to extract sentiment from the tweets which we pull from Twitter. This model is trained using the sentiment140 dataset which is created by Stanford University and contains over 1 million labelled tweets that can be used to train a machine learning model to conduct sentiment analysis [7, 9, 12]. Even Though this dataset is a good starting point, it is not a security specific dataset and contains general tweets taken from Twitter. Because of this, we take the security specific tweets which we scrape from Twitter and make our own dataset which is gradually added into the current dataset. We use the updated dataset to periodically re-train the machine learning model making it more specific to our domain. That is beneficial to produce a more robust dataset that is more tailored to our needs and ultimately provides a better classification of tweets. However, it is also beneficial in providing a security tweet specific dataset which can be used in the future to help do new research and train other models.

Twitter produces over 350,000 tweets per minute which creates noise considering we are only interested in security related tweets. Because of this, the tweets we scrape are filtered based on security relevant accounts which in the past have provided relevant information on attacks, vulnerabilities or exploits. These tweets are then filtered once again based on whether they contain certain security words and company names. Data cleaning is performed on the tweets to get them into a useable format and also to adjust the timestamps based on local time.

This research makes the following contributions to the existing knowledge base. Firstly we aim to expand and enhance current understanding by using real time user generated field specific data in the form of security tweets. From this, we have a data tree based on security words and companies that includes sentiment towards those entities at a given point. This information can then be presented to an analyst for investigation or to a device such as a firewall to influence its processes. Secondly, we aim to create a security tweet specific dataset to be used in this framework as training data and for future research.

Paper Organization - Section 2 will look at related work in the area and analyze similar works and where this study fits in the overall landscape. Section 3 will look at the System Architecture and the individual modules that encompass the system. In Section 4 we go over our provisional testing method for this framework, and in Section 5 we evaluate our testing method and its outcomes. Section 6 goes into the ways we will expand this work in the future, Section 7 explains the threats to validity of this research, and Finally, Section 8 explains our conclusions.

2 RELATED WORK

In Mittal et al. [18] they present CyberTwitter, a framework that analyzes tweets and also outputs alerts that can be used for analysts or other systems. This system uses a Security Vulnerability Concept Extractor to filter based on terms related to security vulnerabilities and then uses a cyber security knowledge base to map strings to real world conceptual instances. That differs to our system in the way tweets are processed and analyzed. Our work currently uses sentiment analysis as the primary data point to base our results

	Category
0	Polarity
1	Tweet ID
2	Date and Time
3	Query
4	User name
5	Tweet text

Table 1: Fields contained in sentiment140 dataset and that we are retrieving from Twitter

Figure 2: Result after performing data cleaning on the original tweet

off of and has the added aspect of creating and updating a security dataset to continue training our classification model.

In Hernandez-Suarez et al. [13] they aim to predict cyber attacks using social sentiment analysis of tweets. That is done by scraping tweets and putting them through a machine learning model which classifies tweets as positive, negative or security. The number of security or positive/negative tweets are then compared to the dates of large events during the 2016 US presidential campaign. This study measured the total number of tweets and then correlated it to events like cyber attacks to show that it is possible to predict attacks on a monthly basis. Our study is more focused on providing real time security information which can be useful for both analysts and other systems and devices.

In Kawakita and Shima [17] they devised a tool which collects data from multiple sites, blogs, and social media, it then uses automated analytics to make a prediction based on that data and outputs that information in STIX format for easier dissemination. There method implemented an algorithm based on Moving Average Convergence Divergence which is a technique used in Financial engineering for predicting stock market changes, using this method they were able to get a 56.1% accuracy in detecting cyber attacks earlier.

3 SYSTEM ARCHITECTURE

This system consists of four componenents 1. Tweet mining, 2. Datasets, 3. Classification and 4. Output. In the tweet mining phase, we use the python API Tweepy to scrape tweets that have been posted in a determined period of time. These tweets are cleaned to remove Twitter specific commands which don't add to our purposes; they are filtered based on security related accounts and security words, and then inputted into the classification model. The classification model is trained using the sentiment140 dataset [9] and predicts the sentiment of the existing security tweets which we scraped. In parallel, the security specific tweets that are being scraped are saved in their own dataset and periodically added to the existing dataset to create a security tweet specific dataset to achieve better classification by having more specific training data. Once the cleaned and filtered data is put through the model it is classified with a positive or negative sentiment, and provided to the end subject whether that be an analyst or another device.

3.1 Tweet Mining

Twitter is an online social networking site that allows account holding users to publish short 140 character messages. We will use this platform to collect user generated real time tweets. The real time nature of this site benefits us since the importance of time in cyber attacks is so crucial. Even if one company or organization is successfully attacked the rapid dissemination of security data can help others save themselves from the attack. By using text mining, we are able to quickly gather and prepare data using coded algorithms. That is something that in the past would either take an extended amount of time or was just not possible. Today by leveraging these techniques we can gather the data from users tweets and process them into a form that can then be easily used for other components.

The first step in our framework is to scrape tweets from Twitter. That is performed using the Python API Tweepy which provides access to the Twitter RESTful API methods. Using this API, we can gain access to Twitter using authorization tokens. Once we have access to Twitter, we use the "user_timeline" or "Cursor" methods to access tweets by searching keywords and user accounts timelines. At this point, we can also specify other parameters such as only retrieving tweets in a certain language and the number of tweets to retrieve. In our case we only retrieve English tweets since our model is trained on English data. When fetching tweets, Tweepy returns a status object which contains various fields including the the fields that can be seen in Table 1. From this object, we take the data that we use to store in the security tweet dataset and also to input into the classification model.

At this moment we are concerned with taking advantage of the real time nature of Twitter and providing information that is current. Because of this, we look at tweets in a one hour timeframe. The timestamp data retrieved from the status's object created_at field is changed to coincide with local time. This way we can make sure the tweets we take from the last hour from all around the world are in sync. Once we have taken the tweets using Tweepy, we clean the data to have it in a format that is easy to process.

Data Cleaning: Once we have mined our data, some processing and data cleaning still needs to be performed to get that data into a state that can be easily used in the other components of the framework. When the tweets are retrieved, due to the nature of Twitter they can contain many different types of data, such as links, @mention's, unicode, retweets, etc. For sentiment analysis, much of this data does not contain great value. Because of this, we remove links and @mention's. In the case of data being coded in an unreadable standard we decode it to present its actual meaning and finally for negative contractions such as "isn't" or "aren't" we expand these terms into their full forms, e.g, "is not", "are not". We do not remove hashtags # since they can often contain information which is useful for classification and filtering purposes. In Figure 2 we can

see an example of a typical tweet on the site www.twitter.com and the subsequent output once data cleaning has been performed on that particular tweet.

Data Filtering: In this study, we are concerned with analyzing tweets that have cyber security relevance, not all tweets on all subjects. Because of this we first filter the tweets we scrape by user account. In Hernandez-Suarez et al. [13] well identified Twitter accounts related to Hacktivists, cyber-security feeds, researchers, enthusiasts, and companies are identified. We use this list as the basis for our account filtering and also add to this list with other well known cyber security related accounts which can be seen in Table 2. An alternative approach would be to scan all tweets posted every hour on Twitter but since there are approximately 20 million tweets posted per hour the processing power required is not feasible for this project. Also by scanning all the accounts, there would be much noise which could harm the results. Because of this, we have decided to filter by well known security accounts that have some history of being credible.

Once we have filtered the tweets we scraped by account we then filter these tweets again based on security words. That is done because even though we have first filtered by security related accounts, all these accounts are not always guaranteed to post purely security related content. Since we want to analyze tweets that are related to security subjects only, we filter these tweets once again by relevant security words. The list of security words used and security accounts are contained in Table 2.

Finally, we also extract Company or Organization names from the tweets to look into the live sentiment. In previous attacks such as the attack carried out by lulzsec on Sony or the various attacks done by Anonymous, we have seen that on some occasions there have been messages posted to Twitter showing negative sentiment towards the victim and even threats of attacks. By correlating the current sentiment for specific companies or organizations with the amount of negative tweets that contain security words for the same company, we may be able to see a trend of an imminent attack on a particular company and start preparing for it.

3.2 Datasets

For this study, we required training data for our sentiment classification model, this data needed to contain a corresponding sentiment label of positive or negative. No existing tweet dataset related explicitly to cyber security; because of this, we chose to use the sentiment140 dataset made by Stanford University [9]. The data set consists of 6 fields which are shown in Table 1. For this study, we are only making use of the text field for training and the polarity of that text as the label. The dataset contains over 1.5 million entries with half of the total being positive and half negative entries. As stated previously we also had to do data cleaning to put it in the same format as the text data we were scraping.

Utilizing this dataset proved to be a good starting point for training our model allowing us to get a classification model working, even so, we felt that having a dataset that was specific to the particular data we were working with could return better results and also be useful for future work in the field. Because of this, with the aim of creating a dataset specific to security related tweets, we created a

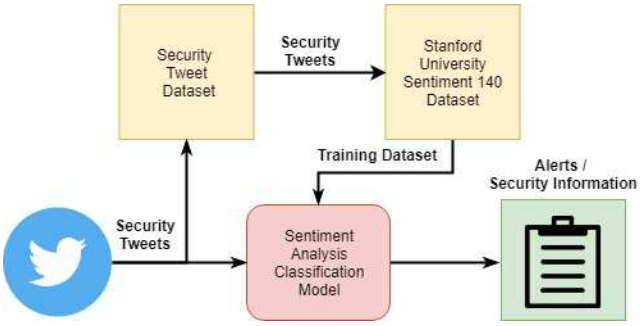

Figure 3: Diagram of overall system architecture

system that allows us to update our model on new security related data at regular intervals.

That was implemented by using the existing tweet scraping system but not filtering by security account and only using security word filtering. After a predetermined amount of time, these tweets were added to the existing dataset which we use to retrain the classification model. By incorporating the new dataset, we hope to achieve better classification since the dataset is more closely suited to the real data we are using, as well as giving us a security specific dataset which we can use for training and/or in future work.

This method is also useful because the security industry has new terms and names for attacks being created on a regular basis, for instance the term wannacry refering to the ransomware attack was not considered security related until after the attack was released. This is the case with many other attack names and by scraping tweets with security words there is a very good chance that new attack names will end up in the dataset by association. For this reason also it is beneficial to continue updating our dataset on a regular basis to contain these new terms.

3.3 Classification

There are multiple types of classification methods for sentiment analysis; two popular methods are the lexicon based method and the machine learning method. In the lexicon based method, dictionaries are employed that contain semantic orientation and strength. From this dictionary, we can calculate a score which will tell us whether the text is positive or negative. This method assumes that the polarity of a text comes from the sum of the individual words and their weights [6]. This method can have high precision but low recall meaning we can get more false negatives, but it has less false positives. Machine learning approaches work by training a classification model using a training set. The data used to train this model can have labeled examples meaning that it has examples of texts that have already been classified as positive or negative. We can then use this data to train our model and input new unseen examples into the model to be classified. For our system, we have chosen to use the machine learning method because we believe it is more promising for the future since data sources are growing at such a fast pace.

To construct our classifier, we are using the scikit learn library to create the model and pandas for data structures. There are various types of machine learning classifiers that can be used like Naive

Security Accounts
"Onion_ID","CSOonline","TheHackersNews","threatpost", "securityaffairs","TripwireInc","deb_infosec","jaysonstreet", "WaPoAnon","AnonymousPress","observingsentin","freeanons", "Global_hackers","AnonymousVideo","Bitdefender", "Silensec","Malwarebytes","NakedSecurity","kaspersky", "NortonOnline","WHNSC","Peerlyst","mikko", "briankrebs","neiljrubenking","dangoodin001","gcluley", "campuscodi","peterkruse","e_kaspersky","troyhunt", "SwiftOnSecurity"
Security Words
"ddos","phishing","botnet","dos","xss","smb", "wannacry","heartbleed","ransomware","trojan", "spyware","exploit","virus","malware","mitm", "brute force","petya","mirai","stuxnet", "eternalblue","anonymous"

Table 2: Table of security accounts and words used for filtering

Bayes Classifier, Support Vector Machine or Multi Layer Perceptron. All these classifiers have their benefits and disadvantages, but for this system, we use a logistic regression model. Logistic regression is a supervised classification algorithm which gives discrete values for its output based on a set of inputs. Logistic regression is a discrimininative classifier which is used for modelling the dependence of unobserved values on observed values. It uses a logistic function to get the likelihood directly and also covers binary dependent variables [10].

Before inputting our text data, we first used TfidfVectorizor which replaces both the function of CountVectorizor and TfidfTransformer. Using this we can convert our text data into a matrix of token counts and then transform this matrix into a normalized tfidf representation that puts weights on the terms and is used for document classification. The point of this is to represent the impact of tokens by occurence and uniqueness more accurately. Along with this, we used Pipeline to combine the two steps to be able to use a single object. Once the text data is transformed, it can be used to train our logistic regression model. For training our classifier, we use 80% of the dataset for training and 20% for testing. On our testing set, we are currently achieving 80.2% accuracy.

3.4 Output Information

Once our model gives us a classification for our data, we can produce security alert information based on the sentiment data and the general security tweet data. In this case, we are showing some simple uses that can be useful for analysts to judge the real time security atmosphere based on this data. That could be especially useful to smaller companies which do not have access to large scale security suites and can prove to be another resource to help make an informed decision.

Firstly we output a live representation of the Overall Security Sentiment based on tweets. That is based on the percentage of negative tweets being posted in the last hour. That can help to show the general public attitude at one time.

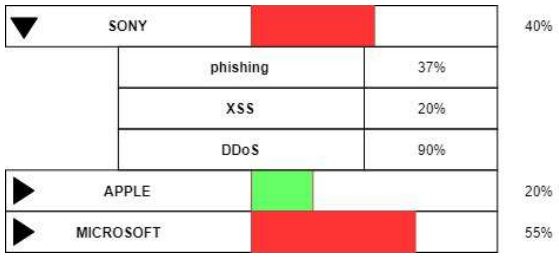

Figure 4: Representation of output for company aspect of alert

We also show overall sentiment for companies based on tweets that contain company names, we scan scraped tweets for certain company names and show whether the current public sentiment on Twitter for that company is positive or negative and by what percentage. If there is a very high negative sentiment towards a particular company it could be wise to consider being more alert for attacks especially to public facing entities like web servers that can be defaced or DDoS'ed. To help pinpoint this further, we also show which security terms are most prevalent in the tweets which contain that companies name. That can help the analyst be more prepared and alert for a specific types of attacks. For example, if an analyst can see that general sentiment towards their company is very low, and terms such as DDoS are very high in their tweets and contain negative sentiment also, it would be astute to pay closer attention to parts of the network which could be affected by a DDoS attack. This information can also be automatically provided to a security device such as a firewall which which can change rules based on the same information.

4 TESTING

We did provisional testing of our security tweet dataset for a period of five days from 10:30 am Sunday 5th of August 2018 until 11:43 am Thursday 9th of August 2018. During this time we had the dataset creation module of our research scrapping tweets in 1-hour intervals and filtering them using our security word list consisting of terms such as "ddos", "phishing", "xss" and specific vulnerability names like "wannacry" and "petya". It saved tweets in a data store based on five fields which can be seen in Table 1. These fields coincide with the sentiment140 dataset to allow for better comparison. During this phase, we were able to scrape 97,112 tweets, with 31,451 tweets recorded with sentiment four relating to positive, and 65,660 tweets recorded with the sentiment zero relating to negative. In Figure 8 we can see the top 100 words that appear at least 800 times in the tweet texts used in the security tweet dataset not including stop words. The size of the words pertains to its frequency meaning a term such as Petya appeared many more times than hit. In Table 3 we can see the top 10 words by frequency, as may be expected many of these words are contained in our security words list but some such as TSMC and uiwix are security specific words which are not in our list and were added to the dataset by association.

During the other testing phase, we tested the main flow of the system by scraping tweets and subsequently classifying those tweets

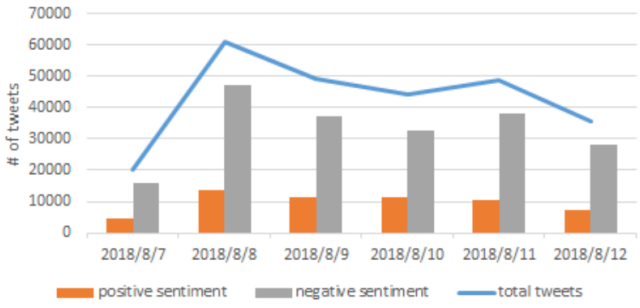

Figure 5: Total security tweets filtered by security account

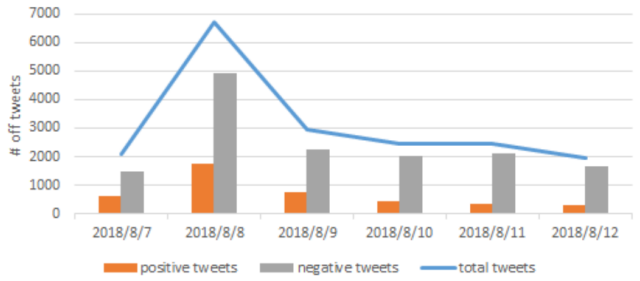

Figure 6: Total security tweets filtered by security account and security words

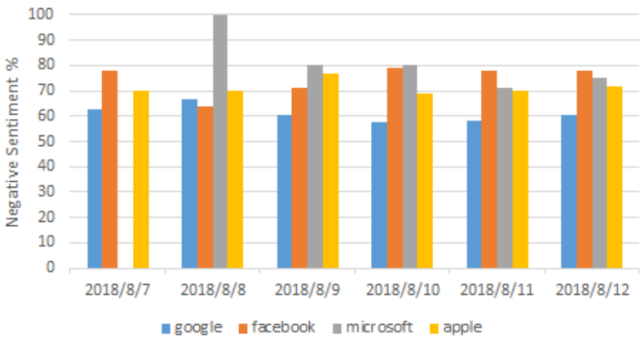

Figure 7: Sentiment analysis by company, filtered by security account

and producing security information based on the information gathered. We scraped tweets at 15-minute intervals based on the filtering techniques described in the Tweet Mining section. We stored our output data as multiple log files which gave us various statistics on the data. This included the number of total tweets and overall sentiment analysis, quantities of positive and negative tweets, sentiment analysis based on company names, and the number of positive or negative tweets related to each company. The total number of tweets scraped was 259,192 with 18,658 of them containing security words. The day with the most activity was the 8th of August with 35.9% of total tweets coming on this day. In Figure 5 and 6 we can

Keyword	Frequency
petya	7090
attack	6777
wannacry	6458
ransomware	5825
phishing	5174
dos	4874
malware	4815
attacks	4169
injection	3828
security	3767
TSMC	2495
uiwix	1361

Table 3: Words in security tweet dataset by frequency

see that the amount of negative tweets outnumbers positive tweets with them taking up 70 to 85% of the tweets on a daily basis.

In Figure 7 we can see the sentiment analysis conducted on the scraped tweets based on company name. When a company name in our company list appeared in one of the scraped tweets, we would record the sentiment of that tweet and tally it towards the current sentiment of that company. Figure 7 shows the sentiment analysis for four companies across six days; all companies received overall negative sentiment, this is not surprising when we see the amount of overall negative tweets were so high. We also recorded the tweets that contained company names, security words and were filtered by security accounts. Somewhat surprisingly a tiny amount of tweets came from a security account and contained security words and company names. Apple was the highest having five tweets in total and Microsoft having two. That is most likely due to the fact there were no major cyber security events for those companies on those days.

5 EVALUATION

Evaluating our dataset comes in two aspects, the quality of the dataset and also the usefulness of the dataset as a whole. When looking at the quality of data contained in the dataset and particularly in the tweet text, we can see some promising aspects and also things to improve upon. Firstly when we look at Figure 8 which is the wordcloud based on our tweet data from the dataset we can see that we have created a more specific security tweet dataset compared to the sentiment140 dataset. This dataset can be used to retrain our classifier model to provide better classification and also used for other types of research into security sentiment analysis or social networking. Even so, there are some components which can be improved that will make the dataset more useful. Firstly, we use various security terms in our filtering process such as Petya which is a ransomware attack and SMB which is an access protocol that is often attacked. From looking at the dataset we can see many entries with these terms which do not relate to security, this is due to the fact that Petya is a name in the slavic region and SMB is an acronym for a sports team. That has made it that some of the entries are not security related. To remedy this, we can use a second pass system for the words that are known to have meanings outside

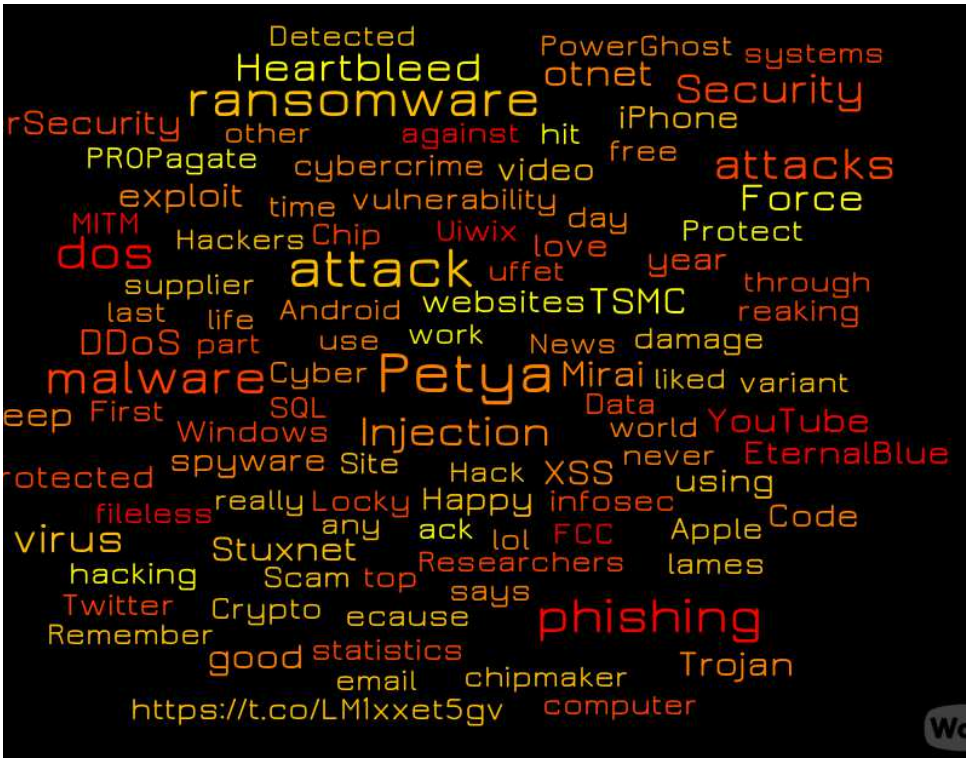

Figure 8: Word cloud of terms contained in security tweet dataset by frequency

of the security field. For these words, we could go over the tweet a second time looking for much broader terms related to security such as "attack", "hack" or "protect". By doing this, we should be able to determine whether or not the tweet is security related more accurately.

Another aspect that we identified as a benefit to creating our own security tweet related dataset was that by association new words and names of attacks that are not already in the dataset will be automatically added. This was shown to be the case, in Table 3 we can see that the last entry is called uiwix and was found 1361 times. This term was not in the original list of security words but was found to occur quite frequently by association with other security terms. Uiwix is a newer type of ransomware that uses SMB vulnerabilities used by WannaCry ransomeware [14]. The fact that this term was not in our original security word filter but still came into our dataset shows that this method allows for new attacks and tools to be integrated and analyzed. That is beneficial because many times analysts will not have the time to go through vast amounts of data individually but can use this type of resource to find new potential threats.

Similarly, in Table 3 we can see the term TSMC which is not in any of our filters but has a high frequency in the dataset. This term stands for Taiwan Semiconductor Manufacturing Company which was attacked by a mutated version of a popular ransomware called wannacry which shut down various factories and can possibly cost the company hundreds of millions of dollars [19].

Unfortunately, we were not running our tests during the time of the attack, but most likely it would have shown both TSMC and associatevely ransomware and wannacry as a much larger proportion of the dataset. Our company filtering function is useful in this type of scenario. Unfortunately, in our tests we were not filtering by this company name, but in the case that we were, it is likely that we would have had a high number of tweets for that company containing security words. Much of that information would have been coming after the attack but even if some came before it could given an opportunity for the company to be slightly more alert and prepared.

6 FUTURE WORK

In this paper, we presented a framework that performs sentiment analysis on security based tweets with the aim to provide relevant security information that can be used by analysts or by security devices. As a byproduct of this system, we also show an updating security tweet dataset which we use to retrain our classification model on a regular basis to achieve better results.

We will continue this work by adding other data sources such as security blogs, security sites, dark web sites and other social media. By doing this, we can extend our range and produce a more accurate representation of the general sentiment in the security community.

To enhance the dataset we are creating we will add a more granulated sentiment analysis system rather than a positive or negative classification as well as test our dataset using other types

of classification models with the aim of improving classification accuracy and hence returning better results.

Finally we will endeavour to combine the output data we have from this system with real time network traffic data to help defend from certain types of cyber attacks.

This future work is all done with the motivation of creating a system that can help provide relevant security information to analysts and also help proactively and dynamically counter against cyber threats.

7 THREATS TO VALIDITY

External Validity: In this study, we use Twitter as our data source which means that our data is restricted to people that have accounts and use Twitter. That can affect the application of our conclusions externally since it can be argued that Twitter alone is not a complete representation of the broader ecosystem since there may be countries and companies that choose not to use Twitter and prefer other mediums. Even though it is correct that there are places which may prefer to use blogs or other mediums to disseminate their information various previous attacks which were predicted and warned on Twitter show that there is value in using Twitter for this research. Having said that it would be beneficial to add other data sources like blogs and websites into this system to produce an even greater overview.

Internal Validity: For this particular system we used a filtering method which only scrapes tweets based on a list of known security accounts. That was partially done due to processing restrictions on the number of tweets that are published on a daily or even hourly basis, and also to reduce noise which can cause false positives. Because of this we restricted the tweets we scrape to well known security related accounts, this narrows our view from a larger group and restricts our view to people only in the cyber security field.

Construct Validity: In this study, we are going off the construct that sentiment analysis can provide useful information that can help proactively counter attacks. There are previous systems that use sentiment analysis and other methods to help predict cyber attacks with varying success. At this point this study can provide alerts and information based on sentiment analysis to someone like an analyst which can help them make a decision about the threat level. At this point, it is not capable of concretely setting a parameter which can detect an attack.

8 CONCLUSION

In this paper, we presented a framework that performs text mining and sentiment analysis on security based tweets with the aim to provide cyber security field relevant information. This is done by, scraping Twitter for security relevant tweets which we filter based on security accounts and words. We then analyze and also conduct sentiment analysis on these tweets using a machine learning model. Moreover we also take the security word specific tweets which we scrape from Twitter and make our own dataset utilizing these tweets.

With the results we have obtained from our testing phase we have shown that using this method we are able to obtain a security specific tweet dataset that is able to incorporate new cyber security specific terms. We have also shown a method to process and classify real time user generated tweets to produce security information. This information can then be presented to an analyst for investigation or to a device such as a firewall to influence its processes.

REFERENCES

[1] Omnicore Agency. 2018. Twitter by the Numbers: Stats, Demographics & Fun Facts. (2018). https://www.omnicoreagency.com/twitter-statistics/

[2] Pablo Aragón, Karolin Eva Kappler, Andreas Kaltenbrunner, David Laniado, and Yana Volkovich. 2013. Communication dynamics in twitter during political campaigns: The case of the 2011 Spanish national election. *Policy & Internet* 5, 2 (2013), 183–206.

[3] BBC. 2015. Hackers Anonymous disable extremist social media accounts. (2015). http://www.bbc.co.uk/newsbeat/article/31313610/hackers-anonymous-disable-extremist-social-media-accounts

[4] Gillian Cleary, Mayee Corpin, Orla Cox, Hon Lau, Benjamin Nahorney, Dick OâĂŹBrien, Brigid OâĂŹGorman, John-Paul Power, Scott Wallace, et al. 2018. *2018 Internet Security Threat Report.* Technical Report. SYMANTEC CORPORATION.

[5] Linh Dang-Xuan, Stefan Stieglitz, Jennifer Wladarsch, and Christoph Neuberger. 2013. An investigation of influentials and the role of sentiment in political communication on Twitter during election periods. *Information, Communication & Society* 16, 5 (2013), 795–825.

[6] MD Devika, C ª Sunitha, and Amal Ganesh. 2016. Sentiment analysis: A comparative study on different approaches. *Procedia Computer Science* 87 (2016), 44–49.

[7] Natalie Friedrich, Timothy D Bowman, Wolfgang G Stock, and Stefanie Haustein. 2015. Adapting sentiment analysis for tweets linking to scientific papers. *arXiv preprint arXiv:1507.01967* (2015).

[8] Robin Gandhi, Anup Sharma, William Mahoney, William Sousan, Qiuming Zhu, and Phillip Laplante. 2011. Dimensions of cyber-attacks: Cultural, social, economic, and political. *IEEE Technology and Society Magazine* 30, 1 (2011), 28–38.

[9] Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1, 12 (2009).

[10] Alec Go, Lei Huang, and Richa Bhayani. 2009. Twitter sentiment analysis. *Entropy* 17 (2009), 252.

[11] Abby Goodrum and Mark Manion. 2000. The ethics of hacktivism. *Journal of information ethics* 9, 2 (2000), 51.

[12] A Gupta, P Kumaraguru, C Castillo, and P Meier. 2014. TweetCred: A Real-time Web-based System for Assessing Credibility of Content on Twitter In: Proc. In *6th International Conference on Social Informatics (SocInfo). Barcelona, Spain.*

[13] Aldo Hernandez-Suarez, Gabriel Sanchez-Perez, Karina Toscano-Medina, Victor Martinez-Hernandez, Hector Perez-Meana, Jesus Olivares-Mercado, and Victor Sanchez. 2018. Social Sentiment Sensor in Twitter for Predicting Cyber-Attacks Using âĎŞ1 Regularization. *Sensors (Basel, Switzerland)* 18, 5 (2018).

[14] Trend Micro Incorporated. 2017. After WannaCry, UIWIX Ransomware and Monero-Mining Malware Follow Suit. (2017). https://blog.trendmicro.com/trendlabs-security-intelligence/wannacry-uiwix-ransomware-monero-mining-malware-follow-suit/

[15] Adam G Klein. 2015. Vigilante media: unveiling Anonymous and the hacktivist persona in the global press. *Communication Monographs* 82, 3 (2015), 379–401.

[16] Robert Mackey. 2010. àĂŸOperation PaybackâĂŹattacks target MasterCard and PayPal sites to avenge WikiLeaks. *New York Times* (2010).

[17] Kawakita Masaru and Shima Shigeyoshi. 2018. *Detection, Auto Analysis of Cyber Threats Using Open Source Intelligence.* Technical Report 12. Security Research Laboratories, NEC.

[18] Sudip Mittal, Prajit Kumar Das, Varish Mulwad, Anupam Joshi, and Tim Finin. 2016. Cybertwitter: Using twitter to generate alerts for cybersecurity threats and vulnerabilities. In *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE Press, 860–867.

[19] The Hacker News. 2018. TSMC Chip Maker Blames WannaCry Malware for Production Halt. (2018). https://thehackernews.com/2018/08/tsmc-wannacry-ransomware-attack.html

[20] Brent R Rowe and Michael P Gallaher. 2006. Private sector cyber security investment strategies: An empirical analysis. In *The fifth workshop on the economics of information security (WEIS06).*

[21] Bruno Takahashi, Edson C Tandoc Jr, and Christine Carmichael. 2015. Communicating on Twitter during a disaster: An analysis of tweets during Typhoon Haiyan in the Philippines. *Computers in Human Behavior* 50 (2015), 392–398.

[22] Peter Woollacott. 2015. Threat Information Overload is Overwhelming Security Analysts. (2015). https://www.channelfutures.com/security/threat-information-overload-overwhelming-security-analysts