IEEE Transaction on Machine Learning,Volume:11,Issue Date:16.June.2023

Automated Emerging Cyber Threat Identification and Profiling Based on Natural Language Processing

RENATO MARINHO AND RAIMIR HOLANDA
Graduate Program in Applied Informatics, University of Fortaleza, Fortaleza 60811-905, Brazil
Morphus Labs, Fortaleza 60811-908, Brazil

Corresponding author: Renato Marinho (rmarinho@morphus.com.br)

This work was supported by Morphus Labs.

ABSTRACT The time window between the disclosure of a new cyber vulnerability and its use by cybercriminals has been getting smaller and smaller over time. Recent episodes, such as Log4j vulnerability, exemplifies this well. Within hours after the exploit being released, attackers started scanning the internet looking for vulnerable hosts to deploy threats like cryptocurrency miners and ransomware on vulnerable systems. Thus, it becomes imperative for the cybersecurity defense strategy to detect threats and their capabilities as early as possible to maximize the success of prevention actions. Although crucial, discovering new threats is a challenging activity for security analysts due to the immense volume of data and information sources to be analyzed for signs that a threat is emerging. In this sense, we present a framework for automatic identification and profiling of emerging threats using Twitter messages as a source of events and MITRE ATT&CK as a source of knowledge for threat characterization. The framework comprises three main parts: identification of cyber threats and their names; profiling the identified threat in terms of its intentions or goals by employing two machine learning layers to filter and classify tweets; and alarm generation based on the threat's risk. The main contribution of our work is the approach to characterize or profile the identified threats in terms of their intentions or goals, providing additional context on the threat and avenues for mitigation. In our experiments, the profiling stage reached an F1 score of 77% in correctly profiling discovered threats.

INDEX TERMS Cyber threat discovery, cyber threat profiling, emerging threats, machine learning, NLP, OSINT.

I. INTRODUCTION

Recently there has been an increasing reliance on the Internet for business, government, and social interactions as a result of a trend of hyper-connectivity and hyper-mobility. While the Internet has become an indispensable infrastructure for businesses, governments, and societies, there is also an increased risk of cyber attacks with different motivations and intentions. Preventing organizations from cyber exploits needs timely intelligence about cyber vulnerabilities and attacks, referred to as threats [1].

Threat intelligence is defined as "evidence-based knowledge, including context, mechanisms, indicators, implications, and actionable advice, about an existing or emerging menace or hazard to assets that can be used to inform decisions regarding the subject's response to that menace or hazard" [2]. Threat intelligence in cyber security domain, or cyber threat intelligence, provides timely and relevant information, such as signatures of the attacks, that can help reduce the uncertainty in identifying potential security vulnerabilities and attacks.

Cyber threat intelligence can generally be extracted from informal or formal sources, which officially release threat information in structured data format. Structured threat intelligence adheres to a well-defined data model, with a common format and structure. Structured cyber threat intelligence, therefore, can be easily parsed by security tools to analyze and respond to security threats accordingly. Examples of formal

The associate editor coordinating the review of this manuscript and approving it for publication was Agostino Forestiero.

sources of cyber threat intelligence include the Common Vulnerabilities and Exposures (CVE) database[1] and the National Vulnerability Database (NVD).[2]

Cyber threat intelligence is also available on informal sources, such as public blogs, dark webs, forums, and social media platforms. Informal sources allow any person or entity on the Internet to publish, in real-time, the threat information in natural language, or unstructured data format. The unstructured and publicly available threat intelligence is also called Open Source Intelligence (OSINT) [3]. Cyber security-related OSINT are early warning sources for cyber security events such as security vulnerability exploits [4].

To conduct a cyber-attack, malicious actors typically have to 1) identify vulnerabilities, 2) acquire the necessary tools and tradecraft to successfully exploit them, 3) choose a target and recruit participants, 4) create or purchase the infrastructure needed, and 5) plan and execute the attack. Other actors— system administrators, security analysts, and even victims— may discuss vulnerabilities or coordinate a response to attacks [5]. These activities are often conducted online through social media, (open and dark) Web forums, and professional blogs, leaving digital traces behind. Collectively, these digital traces provide valuable insights into evolving cyber threats and can signal a pending or developing attack well before the malicious activity is noted on a target system. For example, exploits are discussed on Twitter before they are publicly disclosed [4] and on darkweb forums even before they are discussed on social media [6].

A. TIMELY INTELLIGENCE

Timely intelligence about new threats and how they can affect their victims may be uncovered from publicly available data sources such as forums and social media and may be vital for organizations to prevent cyber incidents or mitigate their impacts.

The global Wannacry campaign affecting thousands of companies in more than 100 countries in 2017[3] is a good example of the importance of being aware of emerging threats. While the malware started spreading on May, 12 of 2017 a large number of messages from cyber experts on the microblog Twitter started mentioning the term 'wannacry' associated with the terms 'ransomware', 'vulnerability', and 'eternalblue'. Those messages if spotted and analyzed were a signal urging people to patch the critical vulnerability MS17-010 as soon as possible to prevent Wannacry to spread over internal organization's networks using EternalBlue exploit.

While it is important, doing this type of early identification is quite challenging. Given the amount of data coming from social media such as Twitter and the required agility to make sense of them timely identifying new threats would not be an easy or even feasible task for humans on a day-to-day basis. It is necessary automation to retrieve, process, and classify the data and to generate alarms for spotted cyber threats.

B. TWITTER AS CYBER INTELLIGENCE SOURCE

Open Source Intelligence (OSINT) is intelligence gathered from public-available sources such as social network sites, forums, wikis, blogs, and so on [7]. Malicious actors, system administrators, security analysts, and victims of cyber attacks usually use such platforms to discuss vulnerabilities, and exploits or to coordinate a response to attacks. Although more difficult to consume due to the volume and unstructured format of the content, data obtained from OSINT sources can complement intelligence obtained from structured intelligence sources, which usually provide malicious IP addresses and hashes, for example, as indicators of compromise (IOCs) that must be monitored or blocked by security platforms.

Among OSINT sources available, we choose Twitter due to its ability to act as a natural aggregator of multiple sources [8] and its big data characteristics: a large volume of data, a highly diverse pool of users, high accessibility, and, mainly, timely production of new content [9]. The popularity of this medium in the cybersecurity community provides an environment for both offensive and defensive practitioners to discuss, report, and advertise timely indicators of vulnerabilities, attacks, malware, and other types of cyber events that are of interest to security analysts.

In the past decade, Twitter has become an important source of intelligence. The real-time nature of information on Twitter has allowed researchers to use the microblog to extract intelligence about different areas such as terrorist attacks [10], earthquakes [11], forest fires [12] and so on. The value of Twitter with regards to security is well-demonstrated by the numerous initial reports of cyber events, examples of which include disclosures of multiple 0-day, user reports on DDoS attacks, and exposure of ransomware campaigns. For example, in June 2017, the global ransomware outbreak of 'Petya/NotPetya' was discussed widely via Twitter before being reported by mainstream media [13].

Another more recent example of cyber threat initially discussed in Twitter was Log4Shell. Log4Shell was the name given to a 0-day exploit to a vulnerability in Log4j2 (CVE-2021-44228), a popular Java logging library. The Log4j2 vulnerability along with a link to the exploit code, which means the code able to take advantage of a vulnerability in an easy way, was disclosed by the profile @P0rZ9 on December 9th, 2021, on Twitter.

Following this post, hundreds of Twitter profiles, including independent researchers and journalists specialized in cyber security, started to post about the vulnerability.

Given this strong and constant presence of the cyber security community in Twitter, over the recent years, the research on Twitter-based OSINT collection has led to the proposal of multiple frameworks [7], [14], [15], [16], [17] for detection and analysis of threat indicators in the Twitter

[1] http://cve.mitre.org/

[2] https://nvd.nist.gov/

[3] https://money.cnn.com/2017/05/12/technology/ransomware-attack-nsa-microsoft/index.html

stream. The shortness of tweets, which nowadays is a text of 280 maximum characters, is considered one of the main challenges when classifying tweets using machine learning algorithms [18]. In contrast with large document corpora, analyzing short documents such as tweets presents some specific semantic challenges towards extracting terms, relationships, patterns, and actionable insights, in general [19].

C. MITRE ATT&CK

Mitre ATT&CK is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community.[4]

According to MITRE ATT&CK Design and Philosophy [20], the first ATT&CK model was created in September 2013 and was primarily focused on the Windows enterprise environment. It was further refined through internal research and development and subsequently publicly released in May 2015 with 96 techniques organized under 9 tactics. Since then, ATT&CK has experienced tremendous growth based on contributions from the cybersecurity community. Several additional ATT&CK models were created based on the methodology used to create the first ATT&CK. The original ATT&CK was expanded in 2017 beyond Windows to include Mac and Linux and has been referred to as ATT&CK for Enterprise. A complementary model called PREATT&CK was published in 2017 to focus on ''left of exploit'' behavior. ATT&CK for Mobile was also published in 2017 to focus on behavior in the mobile-specific domain. ATT&CK for Cloud was published in 2019 as part of Enterprise to describe behavior against cloud environments and services and ATT&CK for ICS was published in 2020 to document behavior against industrial controls systems

Currently, Mitre ATT&CK has three matrices: Enterprise, Mobile, and ICS (Industrial Control System). Each matrix represents the relationship between tactics, techniques, and sub-techniques and contains a curated knowledge base and model for cyber adversary behavior, reflecting the various phases of an adversary's attack life cycle and the platforms they are known to target. At a high level, ATT&CK is a behavioral model that consists of the following core components:

- Tactics, denoting short-term, tactical adversary goals during an attack;
- Techniques, describing the means by which adversaries achieve tactical goals;
- Sub-techniques, describing more specific means by which adversaries achieve tactical goals at a lower level than techniques; and
- Procedures: documented adversary usage of techniques, their procedures, and other metadata.

[4]https://attack.mitre.org

Our research is based on the Enterprise matrix of version 10.1 of MITRE ATT&CK. It consists of 14 tactics and 191 techniques.

To make it clear, let's consider an adversary who intends to compromise a company to steal confidential data hosted on a specific company's servers. To do so, the adversary would have to find a way to enter the company's systems, then move from host to host until reaching the desired server to finally collect and steal the data. Each adversary's move, since entering the company's systems to steal the data, can be mapped to the MITRE ATT&CK knowledge base. For example, entering company's system is related to the 'Initial Access' **tactic** and can be performed by using the 'Valid Credentials' **technique**.

Continuing with this example, in addition to mapping techniques and tactics, MITRE ATT&CK provides a procedural list of how adversaries proceed to execute each technique. In addition to procedures, MITRE ATT&CK provides mitigation and detection procedures for each technique. Mitigations are recommendations of how defenders should apply to reduce the chance of being successfully targeted by the corresponding technique and detection are ways to detect an intrusion using the technique.

As seen, MITRE ATT&CK is an extremely valuable knowledge base for cyber security professionals in general and especially useful for defenders as they can study and plan defense strategies for different malicious tactics and techniques used by real attackers.

But, in this research, we intend to further leverage MITRE ATT&CK as a source of knowledge not just for humans, but for machines as well. Our objective is to take advantage of this ongoing evolving and collaborative knowledge base to train machine learning algorithms that will be used in filtering tweets related to malicious actions as well as in automatically profiling identified cyber threats in terms of their intents.

II. RELATED WORK

Cybersecurity is becoming an ever-increasing concern for most organizations and much research has been developed in this field over the last few years. Inside these organizations, the Security Operations Center (SOC) is the central nervous system that provides the necessary security against cyber threats. However, to be effective, the SOC requires timely and relevant threat intelligence to accurately and properly monitor, maintain, and secure an IT infrastructure. This leads security analysts to strive for threat awareness by collecting and reading various information feeds. However, if done manually, this results in a tedious and extensive task that may result in little knowledge being obtained given the large amounts of irrelevant information. Research has shown that Open Source Intelligence (OSINT) provides useful information to identify emerging cyber threats.

OSINT is the collection, analysis, and use of data from openly available sources for intelligence purposes [21]. Examples of sources for OSINT are public blogs, dark and deep websites, forums, and social media. In such platforms,

any person or entity on the Internet can publish, in real-time, information in natural language related to cyber security, including incidents, new threats, and vulnerabilities. Among the OSINT sources for cyber threat intelligence, we can highlight the social media Twitter as one of the most representative [22]. Cyber security experts, system administrators, and hackers constantly use Twitter to discuss technical details about cyber attacks and share their experiences [4].

Utilization of OSINT to automatically identify cyber threats via social media, forums and other openly available sources using text analytics was proposed in different researches [1], [23], [7], [24], [25], [26], [13], [27] and [28]. However, most proposals focus on identifying important events related to cyber threats or vulnerabilities but do not propose identifying and profiling cyber threats.

Amongst research, [13] proposes an early cyber threat warning system that mines online chatter from cyber actors on social media, security blogs, and dark web forums to identify words that signal potential cyber-attacks. The framework is comprised by two main components: text mining and warning generation. The text mining phase consists on pre-processing the input data to identify potential threat names by discarding ''known'' terms and selecting repeating ''unknown'' among different sources as they potentially can be the name of a new or discovered cyber threat. The second component, warning generation, is responsible for issuing alarms for unknown terms that meet some requirements, like appearing twice in a given period of time. The approach presented in this research uses keyword filtering as the only strategy to identify cyber threat names, which may result in false positives as unknown words may appear in tweets or other content not necessarily related to cyber security. Additionally, this research does not profile the identified cyber threat.

In [26] an identification and classification approach of cyber threat indicators in the Twitter stream is presented. The research proposes a data-driven approach for modeling and classification of tweets using a cascaded Convolutional Neural Network (CNN) architecture to both classify tweets as related or not to cyber security and classify the cyber-related tweets into a fixed listed of cyber threats. The proposed solution includes a pre-processing phase that uses IBM's Watson Natural Language API to identify tweets related to cyber security according to Watson classification results. Additionally, in the pre-processing phase, there is a pre-labeling step performed by simple string matching on the pure tweet text. The threat types considered were: ''vulnerability'', ''DDoS'', ''ransomware'', ''botnet'', ''data leak'', ''zero-day'' and ''general''. Further, the proposed approach uses CNN models trained to classify each tweet as relevant or irrelevant to cyber security. The relevant tweets are passed to a second CNN layer to be classified as one of the 8 different threat types mentioned above. There are important differences of our proposal compared to this one. First, the proposed approach does not name the identified threat. Naming the threat is an important step to cyber threat intelligence as it may allow analysts to identify and mitigate campaigns based on the historic modus operandi employed by a given threat or group. Second, the proposed approach relies on an external component to classify tweets as related or not to cyber security as opposed to our approach that proposes a component to classify tweets using machine learning trained with the evolving knowledge from MITRE ATT&CK. Third, instead of using a keyword match to pre-filter threats and a fixed list of threat types, we present an approach to profile the identified cyber threat to spot in which phase of phases of the cyber kill chain the given threat operates in. This is important for a cyber threat analyst as he or she may employ the necessary mitigation steps depending on the threat profile.

In [1], a framework for automatically gathering cyber threat intelligence from Twitter is presented. The framework utilizes a novelty detection model to classify the tweets as relevant or irrelevant to Cyber threat intelligence. The novelty classifier learns the features of cyber threat intelligence from the threat descriptions in the Common Vulnerabilities and Exposures (CVE) database [5] and classifies a new unseen tweet as normal or abnormal in relation to Cyber threat intelligence. The normal tweets are considered as Cyber threat relevant while the abnormal tweets are considered as Cyber threat-irrelevant. The paper evaluates the framework on a data set created from the tweets collected over a period of twelve months in 2018 from 50 influential Cyber security-related accounts. During the evaluation, the framework achieved the highest performance of 0.643 measured by the F1-score metric for classifying Cyber threat tweets. According to the authors, the proposed approach outperformed several baselines including binary classification models. Also, was analyzed the correctly classified cyber threat tweets and discovered that 81 of them do not contain a CVE identifier. The authors have also found that 34 out of the 81 tweets can be associated with a CVE identifier included in the top 10 most similar CVE descriptions of each tweet. Despite presenting a proposal to distinguish between relevant and irrelevant tweets, the proposal does not address the identification of threats and their intentions. Those are important requirements for Cyber Threat Intelligence in formulating defense strategies against emerging threats.

The tool proposed in [23] collects tweets from a selected subset of accounts using the Twitter streaming API, and then, by using keyword-based filtering, it discards tweets unrelated to the monitored infrastructure assets. To classify and extract information from tweets the paper uses a sequence of two deep neural networks. The first is a binary classifier based on a Convolutional Neural Network (CNN) architecture used for Natural Language Processing (NLP) [29]. It receives tweets that may be referencing an asset from the monitored infrastructure and labels them as either relevant when the tweets contain security-related information, or irrelevant otherwise. Relevant tweets are processed for information extraction by a Named Entity Recognition (NER) model, implemented as a Bidirectional Long Short-Term Memory (BiLSTM) neural

[5]https://cve.mitre.org/

network [30]. This network labels each word in a tweet with one of six entities used to locate relevant information. Furthermore, the authors chose to use the application of deep learning techniques because of its advantages in the NLP domain [31]. Thus, they propose an end-to-end threat intelligence tool that relies on neural networks with no feature engineering. The pipeline is capable of receiving tweets relevant to infrastructure, selecting those which appear to contain relevant information regarding an asset's security, and extracting valuable entities which can be used to issue a security alert. During the evaluation, was established a methodology through which, according to a defined evaluation metric, the authors compared several variations to the deep learning architectures to select a model which provides the best performance. Furthermore, were compared the proposed models to other well-known classifiers and provided a detailed analysis of the results obtained. The evaluation showed that the approach was capable of finding, on average, more than 92% of the relevant tweets, and matching cybersecurity-relevant labels to named entities within these tweets with an average F1-score above 90%. Based on the best models obtained in their experiments, they retrieved tweets where the NER models were capable of extracting relevant entities and performed a brief analysis that demonstrates the timeliness of Twitter as a valuable source for relevant cyber threat awareness. The approach presented in this work proved to be promising in identifying threats focused on a certain group of assets. However, it becomes limited when the objective is to identify and characterize broader emerging threats, not necessarily aimed at a particular target nor vulnerabilities-specific technologies.

In [19] is presented a machine learning and text information extraction approach to detect cyber threat events in Twitter that are novel and developing using an unsupervised machine learning approach. The detected events are ranked based on an importance score by extracting the tweet terms that are characterized as named entities, keywords, or both. The evaluation of the proposal is carried out by comparing the efficiency and detection error rate to a human annotator ground truth. The proposal proved promising in identifying cyber threat events resulting from the grouping of tweets with similar terms with a true positive rate of 75% and a false positive rate of 16.67%. The work also proposes a heuristic to calculate a score and rank the importance of events based on how influential is the profile on Twitter based on the number of its followers. In our approach, we use similar criteria to calculate the importance degree of an alert based on a number of characteristics of the identified cyber threat including but not limited to the number of followers of a given Twitter profile as detailed in III-A12. The proposed approach does not address the identification of threats and their intentions.

The work proposed in [9] presents a Multi-Task Learning (MTL) approach that merges two models into an end-to-end pipeline for cybersecurity-centric Natural Language Understanding (NLU). MTL is an inductive transfer learning mechanism where a model is trained on multiple tasks, leveraging the knowledge acquired for one to boost the performance of the other [32], [33]. According to the authors, recent work in NLP has shown that MTL can often boost the performance of state-of-the-art models [34]. MTL methodologies have shown not only to improve results on tasks that share a common domain but also that by learning multiple related tasks the model improves its generalization capability, greatly reducing chances of overfitting [35]. The proposed MTL cyber threat intelligence pipeline uses Twitter as its OSINT data stream source. The proposed tool receives tweets, through the Twitter API, from a predetermined set of accounts that have been selected based on their likelihood of outputting security-related content about a specified IT infrastructure. The tweets are filtered based on the mention of IT infrastructure assets and then normalized before they are fed to the Deep Neural Networks (DNN) stage. For processing the text, an MTL DNN model forks into two output modules: a binary classifier and a Named Entity Recognizer (NER). Both share character-level and word-level representation layers which can either be a Convolutional Neural Network (CNN) [29] or a type of Recurrent Neural Network (RNN) such as the Long Short-Term Memory (LSTM) [36]. The combined result of the output modules produces a concise artifact reporting a security event, such as a vulnerability disclosure or security update, to issue an alert. The proposed Multi-Task Learning (MTL) approach is an improvement of a previous research [23] that, although achieves similar results regarding named entity recognition and binary classification task, reduces the complexity of the information pipeline and the procedures required for online update of the dataset and model parameters. However, as the previous research, it becomes limited when the objective is to identify and characterize broader emerging threats, not necessarily aimed at a particular target or vulnerabilities-specific technologies.

A system that automatically generates warnings of imminent or current cyber-threats is approached in [5]. The paper introduces a lightweight framework that leverages online social media sensors such as Twitter and darkweb forums, to generate alerts that function as early warnings of cyber threats. The system monitors the social media feeds of a number of prominent security researchers, analysts, and white-hat hackers, scanning for posts (tweets) related to exploits, vulnerabilities, and other relevant cybersecurity topics. Afterward, it applies text mining techniques to identify important terms and remove irrelevant ones. Then, the system verifies whether the terms that were identified during the filtering stage have ever been used in darkweb hacking forums, and eventually reports the volume of mentions as well as the content of posts. Such information might be extremely valuable since mentions that have been found by the algorithm might point to links to stolen credentials as well as threads where a novel vulnerability is discussed along with source codes aiming at exploiting it. The framework relies on a database, updated daily, of posts published on nearly 200 darkweb and deepweb hacking forums and marketplaces [6], [7], [37].

After all, the system generates warnings for the newly discovered terms, along with their frequency of appearance on social media and darkweb, the contents of possible mentions found in darkweb and deepweb, and a collection of words providing semantic context for facilitating situational awareness and interpretation of the warning. The algorithm design allows for the generation of additional warnings over the same time period. This choice is due to the willingness to keep track of the attention around the possible cyber-threat, and in particular to monitor the evolution of darkweb activities related to the discovered terms. The system proposed in this research provides a mechanism to identify current cyber threats by correlating terms that appears in tweets and eventually in posts on darkweb hacking forums. Despite getting tweets from prominent profiles, it is common for there to be posts that mention terms related to cybersecurity in conjunction with unknown terms that are not necessarily talking about a threat. The absence of a filter that maximizes the chances that the tweets are similar to the description of how a threat acts, can cause the false positive rate to increase, as it actually happened. Additionally, in order to provide a context for the identified threat, the proposed alarm system presents the terms that co-occurred with the unknown term (threat), but seeks to indicate the threat's intent.

Recent research has demonstrated the need and importance in the early detection of emergent cyber threats. Using databases originating mainly from open sources (OSINT), such as Twitter, blogs, dark web, deep web, authors have presented works using the most varied techniques of natural language processing, machine learning, and deep learning to improve the effectiveness in detecting attacks. Our work, however, differs from previous work because in addition to promoting the identification of emerging attacks, we are able to aggregate two new features: 1) define the profiling of the attack based on the MITRE framework and 2) present the results of the identification and profiling under a risk perspective.

III. THE PROPOSED SOLUTION

The overall goal of this work is to propose an approach to automatically identify and profile emerging cyber threats based on OSINT (Open Source Intelligence) in order to generate timely alerts to cyber security engineers. To achieve this goal, we propose a solution whose macro steps are listed below.

1) Continuously monitoring and collecting posts from prominent people and companies on Twitter to mine unknown terms related to cyber threats and malicious campaigns;
2) Using Natural Language Processing (NLP) and Machine Learning (ML) to identify those terms most likely to be threat names and discard those least likely;
3) Leveraging MITRE ATT&CK techniques' procedures examples to identify most likely tactic employed by the discovered threat;
4) Generating timely alerts for new or developing threats along with its characterization or goals associated with a risk rate based on how fast the threat is evolving since its identification.

Alarms will be issued for new or evolving threats along with their characterization, goals, and risk rate based on how fast the threat is evolving since its identification.

A. SOLUTION ARCHITECTURE

This section details each aspect of the proposed architecture shown in Figure 1. Each component of the architecture is numerated and detailed in the following subsections accordingly.

1) LOGSTASH TWITTER API

This is the component that allowed us to collect tweets and insert them into a database for further solution steps. The data collection is done by Logstash,[6] an open-source data collection engine with real-time pipeline capabilities. Logstash is one of the three main components of the Elastic Stack.[7] The other two are Elasticsearch and Kibana.

The Logstash capabilities comprise a broad array of input, filter, and output plugins, with many native codecs further simplifying the data ingestion process. One of the plugins is the Logstash Twitter[8] which allows Logstash to ingest events from the Twitter Streaming API.[9]

One of Logstash's Twitter API parameters is named 'follows', which expects a comma-separated list of user IDs, indicating the users whose Tweets should be delivered on the stream [38]. According to the Twitter documentation, for each user specified, the stream will contain:

- Tweets created by the user;
- Tweets which are retweeted by the user;
- Replies to any Tweet created by the user;
- Retweets of any Tweet created by the user;
- Manual replies created without pressing a reply button (e.g. "@twitterapi I agree").

We compiled a list of 73 well-known and reliable Twitter experts who post frequently on issues related to cybersecurity. The list, shown in Table 1, includes international researchers and security analysts associated with security firms, as well as widely-followed white hat hackers. The list can be arbitrarily extended, but it is important to keep in mind that the selected profiles can interfere in the identification of cyber threats. So, include expert, active, and trusted profiles. The strategy we used to select the profiles was choosing profiles followed by experienced and active cyber threat intelligence analysts and entities we know.

[6]https://www.elastic.co/logstash
[7]https://www.elastic.co/elastic-stack
[8]https://www.elastic.co/guide/en/logstash/current/plugins-inputs-twitter.html
[9]https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data

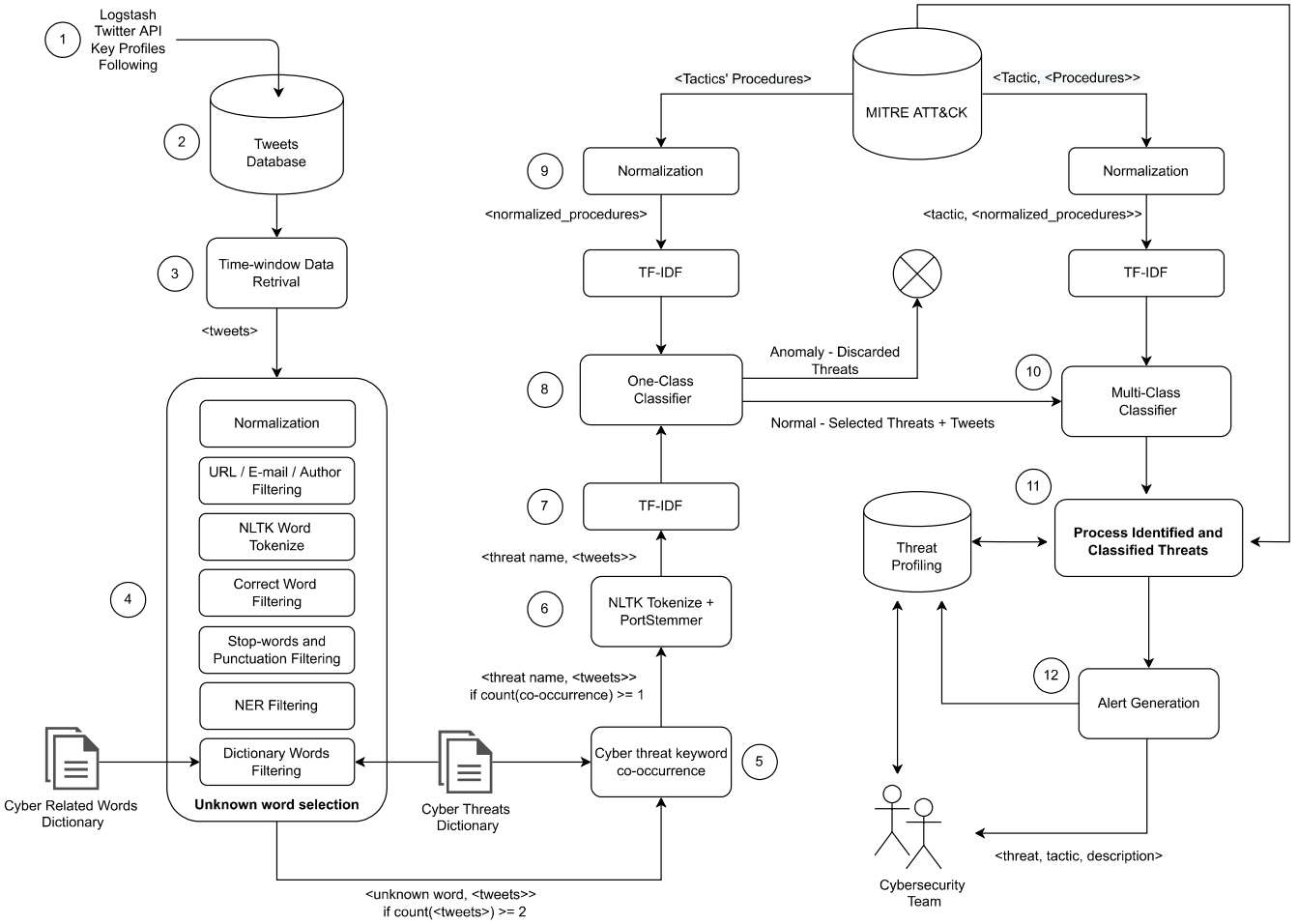

FIGURE 1. **Proposed architecture.**

2) TWEETS DATABASE

The data collected by the Logstash Twitter Plugin is stored in an Elasticsearch Database. Elasticsearch is the distributed search and analytics engine at the heart of the Elastic Stack. It provides near real-time search and analytics for all types of data.[10]

3) TIME-WINDOW DATA RETRIEVAL

The first step of our pipeline consists in collecting Twitter messages from the Tweets Database which were posted within a given time range. Considering that our objective is to provide a continuous threat identification and alerting system, the time range will be a sliding time window considering the end time of the previous time range as the start time for the next time range.

All the resulting Twitter messages will follow to the Unknown Word Selection, described in the next subsection.

[10]https://www.elastic.co/guide/en/elasticsearch/reference/current/elasticsearch-intro.html

4) UNKNOWN WORD SELECTION

The objective of this component is to identify unknown words or terms appearing in collected Twitter messages as they, accordingly to further analysis, may represent the name of the identified emerging threat.

The idea of identifying those unknown terms came from the analysis of cyber threats names, which usually receive very strange names - either given by their creators of by the cyber security experts who first spotted them. Wannacry, NotPetya, Cookthief, Emotet, lokibot, and 16shop are some examples of threat names.

For the proposed architecture, a term is considered unknown if it passes through the Unknown Word Selection pipeline, which comprises the following procedures: Normalization, URL/E-mail/Author filtering, NLTK Word Tokenize, Correct Word Filtering, Stop-words and punctuation filtering, NER (Named Entity Recognition) filtering and, finally, Dictionary words filtering, as described below.

a: NORMALIZATION

Considering that we are using Twitter messages posted by a variety of people and that Twitter itself imposes a length

TABLE 1. Twitter profiles and a respective number of followers.

Profile	Followers	Profile	Followers
0xcharlie	73765	ChrisJohnRiley	21364
DarkReading	261979	Dave_Maynor	15273
ErrataRob	45089	Fox0x01	112202
MarkBaggett	9514	PhysicalDrive0	17766
RobertMLee	46739	SANSDefense	24549
SCMagazine	119341	SecurityWeek	189121
SwiftOnSecurity	329817	WebBreacher	15290
WeldPond	52505	XI_Research	11366
Xylit0l	25144	anton_chuvakin	33556
assolini	12137	attcyber	79842
attritionorg	18858	benjaminwright	1876
chicagoben	9858	codelancer	7008
dangoodin001	39921	dnsstream	1297
drericcole	39680	e_kaspersky	184647
edskoudis	56734	enigma0x3	31738
eric_conrad	10163	halvarflake	31278
harmj0y	36457	hasherezade	53352
haveibeenpwned	130366	iamevltwin	20056
internetofshit	448335	jasonlam_sec	3153
jepayneMSFT	33653	jeremiahg	65277
johullrich	14175	k8em0	103678
malware_traffic	54489	malwareunicorn	151380
metasploit	214231	mikko	211270
mroesch	12282	msftsecresponse	134129
msftsecurity	313236	robtlee	26324
sans_isc	100253	securityweekly	70289
sibertor	20579	taosecurity	59277
teamcymru	41340	thecyberwire	33755
xme	15299	DidierStevens	29951
MalwareTechBlog	246482	SANSEMEA	37066
SANSInstitute	151734	USCERT_gov	161004
alexstamos	80053	hacks4pancakes	136438
lennyzeltser	48801	taviso	118652
threatpost	206793	briankrebs	312214
hdmoore	91804	PhishingAi	9426
TalosSecurity	38464	campuscodi	53680
thezdi	49290		

limit for the post message (nowadays 280 characters), it is very common to have terms for the same meaning written and shortened in different forms. For example, 'C2 server', 'C&C server' are written in different forms but, in the context of cyber security, mean the same thing: 'command and control server'. Command and control servers are computers controlled by an attacker or cybercriminal which is used to send commands to systems compromised by malware and receive stolen data from a target network [39].

By analyzing Twitter posts for the selected profiles for this research, it was possible to identify and implement normalization for 47 terms written in 124 different forms. The term list includes brute-force, data wiping, DDoS, information stealer, drive by, and rootkit. The full list is available at Appendix B.

b: URL, E-MAIL AND AUTHOR FILTERING

The collected Twitter posts retrieved and stored in the Elasticsearch database contain the entire Twitter post. It may include mentions to other users, URLs and e-mail addresses. For the purpose of this solution's component, is to identify unknown words, including those terms that would result in false positives. Thus, those terms are filtered out.

c: NLTK WORD TOKENIZE

This step consists in splitting each collected tweet into words. The process of splitting a sentence into words or just word tokenize is very commonly used by natural language processing solutions. To employ word tokenization into the proposed solution, we use Natural Language ToolKit (NLTK) Python module.[11] The output of this step is, for each tweet, an array of its words or tokens. See in the example below how the content of a tweet is split into tokens:

Tweet: ''The RobbinHood ransomware is using a vulnerable legacy Gigabyte driver in order to get around antivirus protections''.

Tokens: ['The', 'RobbinHood', 'ransomware', 'is', 'using', 'a', 'vulnerable', 'legacy', 'Gigabyte', 'driver', 'in', 'order', 'to', 'get', 'around', 'antivirus', 'protections'].

d: CORRECT WORD FILTERING

This step consists in eliminating words that are correctly written as they are known words as well. This filtering step is implemented by SpellChecker Python module.[12]

e: STOP-WORDS AND PUNCTUATION FILTERING

In this step, English stop words are eliminated. The filtered stop words are those present in Python NLTK 'stopword' English word list corpora, which consists in 179 words including: 'i', 'me', 'myself', 'we', 'our', and so on.

f: NER FILTERING

After applying the above filters in the pipeline, we noticed that, among unknown words, there were many organizations' names like Microsoft, Google, and so on. Although they were really unknown words for the filters used until this point of the pipeline, we should eliminate them because, knowingly, they did not represent threat names.

There is a field called Named Entity Recognition (NER) which is considered a fundamental task in a natural language processing (NLP) system. NER is a subproblem of information extraction and involves processing structured and unstructured data to identify expressions that refer to people, places, organizations, and companies [40]. Thus, applying NER to our pipeline would help reduce the number of companies being considered 'unknown words'.

To take advantage of NER in our solution, we used the project DBpedia.[13] DBpedia is a crowd-sourced community effort to extract structured content from the information created in various Wikipedia projects. This structured information resembles an open knowledge graph (OKG) which is available for everyone on the Web. A knowledge graph is a special kind of database that stores knowledge in a machine-readable form and provides a means for information to be collected, organized, shared, searched, and utilized [41].

[11] https://www.nltk.org/
[12] https://pypi.org/project/pyspellchecker/
[13] https://wiki.dbpedia.org/about

The English version of the DBpedia knowledge base describes 4.58 out of which 4.22 million are classified in a consistent ontology, including 1,445,000 persons, 735,000 places (including 478,000 populated places), 411,000 creative works (including 123,000 music albums, 87,000 films and 19,000 video games), 241,000 organizations (including 58,000 companies and 49,000 educational institutions), 251,000 species and 6,000 diseases.

To implement NER using DBpedia knowledge base we have used the DBpedia Spotlight project.[14] This implementation provides a tool for automatically annotating mentions of DBpedia resources in a given text - which in our case, the collected Twitter messages. For the example in Figure 2, we submitted the text: ''Google is a company based in the United States'' to DBpedia Spotlight. As a result, two entities were recognized: 'Google', an entity classified into different types including ''DBpedia: Company'' with a similarity score of 0.99%; and 'United States' also classified into multiple types including 'DBpedia:country'' with a similarity score of 0.99% as well.


```
[{'URI': 'http://dbpedia.org/resource/Google',
'support': 35031,
'types': 'Wikidata:Q4830453,Wikidata:Q43229,Wikidata:Q24229398,
      DUL:SocialPerson,DUL:Agent,Schema:Organization,
      DBpedia:Organisation,DBpedia:Agent,DBpedia:Company',
'surfaceForm': 'Google',
'offset': 0,
'similarityScore': 0.9995333447365896,
'percentageOfSecondRank': 0.0004260725061849133},
{'URI': 'http://dbpedia.org/resource/United_States',
'support': 551746,
'types': 'Wikidata:Q6256,Schema:Place,Schema:Country,
  DBpedia:PopulatedPlace,DBpedia:Place,DBpedia:Location,
  DBpedia:Country',
'surfaceForm': 'United States',
'offset': 29,
'similarityScore': 0.9998810645002776,
'percentageOfSecondRank': 6.37553219711587e-05}]
```


FIGURE 2. DBPedia query example.

To automate the process of programmatically submitting collected tweets to DBpedia Spotlight, we have used the PySpotlight [15] Python 3 module. PySpotlight is a thin Python wrapper around DBpedia Spotlight's REST Interface. So, for each tweet, all the terms recognized as entities by DBpedia Spotlight were considered known words and, thus, were filtered out in this point of the unknown words selection pipeline.

The filters used, which identify the entity types we were interested in identifying using DBpedia were: DBpedia: Currency, DBpedia: Device, DBpedia:Event, DBpedia: Language, DBpedia:Name, DBpedia:Organisation, DBpedia:Person, DBpedia:Place, DBpedia:ProgrammingLanguage, DBpedia:Software and DBpedia:Website''.

An additional important parameter of DBpedia is confidence. It represents the certainty degree the DBpedia has to attribute a given word to a known entity. The greater the confidence, the smaller the number of entities identified with greater accuracy tends to be. On the other hand, the lower the confidence, the greater the number of entities identified with less accuracy tends to be. In practical terms, the lower the confidence, the greater the number of false positives tends to be.

For this research, we observed that using DBpedia with a high confidence parameter was failing to identify unusual terms as entities, which ended up increasing the number of terms falsely identified as threat names. On the other hand, we couldn't leave it too flexible to the point of DBpedia identifying important words for the context of the tweet as an entity.

Take as an example the following tweet:

SEABORGIUM overlaps with groups tracked as Callisto Group (F-Secure), TA446 (Proofpoint), and COLDRIVER (Google). Security Service of Ukraine (SSU) has associated Callisto with Gamaredon; however, MSTIC has not observed technical intrusion links to support the association.

The following entities would be identified by DBpedia for different confidence parameters:

- Confidence of 0.30: 'Security Service', 'F-Secure', 'Google', 'Proofpoint', 'SSU', 'Ukraine';
- Confidence of 0.40: 'Ukraine', 'F-Secure', 'Proofpoint', 'SSU', 'Google', 'Security
- Confidence of 0.50: 'Google', 'Security Service', 'Ukraine', 'F-Secure'; Service'
- Confidence of 0.60: 'Security Service', 'Google', 'F-Secure', 'Ukraine'.

So, in this example, for the context of our research, where we do not consider entity names as threat names, using a 0.40 confidence, we avoided considering 'Proofpoint' (the name of a cyber security provider), and 'SSU' (Security Service of Ukraine), as cyber threats. Additionally, notice that we did not have a difference in terms of identified entities moving the confidence parameter from 0.30 to 0.40 and from 0.50 to 0.60.

After some experiments, we found that a confidence of 0.40 presented a good balance for our research. This value is slightly lower than the reference value used in the online demo of DBpedia, which is 0.50.[16]

g: DICTIONARY WORDS FILTERING

This is the final step of the unknown words selection pipeline. It consists in filtering out terms that occur in both dictionaries: Cyber Related Words Dictionary and Cyber Threats Dictionary. Both were created by us during the experiments for this research and contain technical terms related to cyber security area. Their content are described below:

- Cyber Related Words Dictionary: this dictionary contains 288 words related to cyber security or technology in general that could be considered 'unknown' to the previous steps of this pipeline but do not represent

[14] https://hub.docker.com/r/dbpedia/dbpedia-spotlight

[15] https://pypi.org/project/pyspotlight/

[16] https://demo.dbpedia-spotlight.org/

threats. Some examples config, vpns, sha1, sha256, urls, dmarc and hackathon. The dictionary is available at Appendix C.

- Cyber Threats Dictionary: this dictionary contains 72 words related to known types of cyber threats or they represent actions usually performed by cyber threats. However, they do not name the threat itself. Some examples: ransomware, trojan, collect, malware, cryptominer, exploit and fileless. The dictionary is available at Appendix A.

5) CYBER THREAT KEYWORD CO-OCCURRENCE

The input for this stage of the solution is a list of unknown words that appear in at least two Twitter posts processed by the previous stage along with the Twitter posts their occurred in. The objective of requiring the unknown word to appear in at least two messages is to avoid unknown terms that were just simply misspelled.

For each unknown word, this stage checks for the occurrence of terms from Cyber Threats Dictionary in Twitter posts the unknown words appear. In other words, we are looking for unknown terms that co-occur with a term related to known types of cyber threats.

The output of this stage are the unknown terms that co-occur with at least one Cyber Threat Dictionary term along with the Twitter messages they appear in.

6) NLTK TOKENIZE AND PORTSTEMMER

For grammatical reasons, documents use different forms of a word, like organize, organizes, and organizing. Additionally, there are families of derivationally related words with similar meanings, such as democracy, democratic, and democratization. But for the purposes of our research, where we measure the co-occurrence of words among different documents, we are mainly interested in measuring it by comparing the root of those words - regardless of the way they were written.

To reduce the words to their root format, we applied Porter's stemming algorithm [42]. The algorithm uses a heuristic process that chops off the end of words. This way, for example, the words organize, organizes and organizing are transformed into 'organ' after applying the stemming algorithm.

This process is applied for each word of each tweet received from the previous step.

The output of this stage, thus, is the unknown terms received from the preceding step along with the respective tweets in stemming format.

7) TF-IDF

In this subsection, we describe the use of TF-IDF (Term Frequency-Inverse Document Frequency) to transform the text documents coming from both MITRE ATT&CK corpus and Twitter messages into a vectorized representation needed by both One-Class and Multi-Class machine learning (ML) algorithms - the next steps in the pipeline.

Machine learning algorithms, more specifically the ones used in this work, operate on a numeric feature space, expecting input as a two-dimensional array where rows are instances and columns feature. To perform ML on the text we need to transform our documents into vector representations such that we can apply numeric machine learning in a process called feature extraction or vectorization [43].

To perform the feature extraction, we employed a method called TF-IDF (Term Frequency-Inverse Document Frequency) [44]. TF-IDF is a numerical representation of the importance (weight) of a term t in a specific document d within a corpus of documents. The weight of term t in a document d is defined as

$$TF - IDF(t, d) = f(t, d) * log(N/n_t), \tag{1}$$

where $f(t, d)$ is the number of the occurrences of term t in document d, N is the total number of documents in the corpus and n_t is the number of the documents containing term t.

In other words, TF-IDF is a numerical measure that represents the importance of a particular word to a specific document within a corpus of documents. Words with high TF-IDF values imply a strong relationship with the document they appear in. The goal of using TF-IDF instead of the raw frequencies of occurrence of a token or word in a given document is to scale down the impact of tokens that occur very frequently in a given corpus and that are hence empirically less informative than features that occur in a small fraction of the training corpus.

To implement TF-IDF, we used the class TfidfVectorizer from Sklearn.[17] The method first converts a collection of text documents into a matrix of token counts. This implementation produces a sparse representation of the counts. Later, the method transforms the count matrix into a normalized TF-IDF representation.

One important parameter of the TfidfVectorizer class is called min_df. Min_df is used to inform the algorithm of a threshold to ignore terms when building the vocabulary that has a document frequency strictly lower than it. In other words, min_df is used for removing terms that appear too infrequently in the collection of documents of corpus. For this reason, this value is also called cut-off in the literature. The default parameter for min_df is 1, which means ignoring terms that appear in less than 1 document. Thus, the default setting does not ignore any terms. In the result section, we performed a variation of the min_df parameter from the default and had a slight improvement in the machine learning accuracy.

The TF-IDF is employed in multiple steps of our Proposed Architecture (please refer the Figure 1) as follows:

- In step 9 we employ TF-IDF in the MITRE ATT&CK techniques' threat procedures examples to produce the ML features used in the One-Class Classifier **training phase**. Here, all the procedures examples of all

[17] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

techniques were put together, independently of the tactic it appears in. To implement this step, we used the method *TfidfVectorizer* provided by the Skitlearn Python Library[18];

- In step 7 we employ TF-IDF to transform the Twitter messages to the document-term matrix using the vocabulary and the term frequency (TF) produced in the training phase. The output of this step will be classified by the One-Class Classifier trained using MITRE ATT&CK corpus;
- In step 10 there is another TF-IDF employed using the MITRE ATT&CK corpus. This time, the TF-IDF calculation input is the MITRE ATT&CK techniques' threat procedures examples for each tactic of the knowledge base. This way, each tactic will have its own ML features that will be used to train the Multi-Class Classifier.

8) ONE-CLASS CLASSIFIER

Until this point of the pipeline, all the filters applied aimed to identify emerging cyber threats based on the principle of their names are unknown words. But, despite the efforts of using Cyber Related and Cyber Threats dictionaries and NER to filter out some words that aren't cyber threats - like company names, our experiments showed us that there is always the possibility to have unknown words passing through the pipeline that aren't cyber threats.

To go a step forward, we implemented an approach to select threat names associated with tweet messages whose content is close to the descriptions of malicious actions.

To discover if a tweet's content is close or not to descriptions of malicious actions we implemented the concept of novelty classification using a One-class classifier. It works by comparing the content of an unseen content (tweet) with the training set to decide if the unseen content is normal/similar or abnormal/different. The tweets which are considered normal by the classifier are sent to a buffer that later will be sent to the Multi-Class classifier. The abnormal tweets are discarded. The One-class classifier is trained only with positive samples, which, in our case, consists of the Mitre ATT&CK techniques' threat procedures.

The One-class classifier, as the name implies, has just one class and it was trained with all the procedures of MITRE ATT&CK, regardless the tactic or technique to which it belongs. In the list below there are some examples of procedures.

- APT18 actors leverage legitimate credentials to log into external remote services;
- APT39 has used SQL injection for initial compromise;
- Lokibotś second stage DLL has set a timer using ''timeSetEvent'' to schedule its next execution;
- Grandoreiro can use malicious browser extensions to steal cookies and other user information;
- Avaddon encrypts the victim system using a combination of AES256 and RSA encryption schemes;
- Babuk can stop specific services related to backups;

In Table 2 there are two examples of terms that were considered cyber threats due to its unknown names, but that was discarded by One-class algorithm as tweet contents are abnormal/different from the classifier.

TABLE 2. Samples of discarded tweets by one-class classifier.

Threat Name	Tweet
anime	There are only 5 groups on Twitter Anime avatar Rose Twitter Wearing sunglasses sitting in car Furries Ben Shapiro
prepper	Stealing their food. This concept is nowhere in prepper culture. Instead, it s the opposite preventing people from stealing food.

The implementation of the One-Class classifier was done using OneClassSVM algorithm from SkitLearn[19] configured with the default Radial Basis Function (RBF) Kernel.

9) NORMALIZATION

The goal of this step is to unify words that mean the same thing but that are eventually spelled differently. To this end, we applied the same normalization procedure we applied to tweets on step III-A4.a to MITRE ATT&CK procedures examples.

An example of the need to apply normalization can be seen in the term brute-force, which is spelled differently in the following procedures. The first procedure was extracted from 'Valid Account' technique and the second from 'Brute Force' technique.

- APT41 performed password **brute-force** attacks on the local admin account.
- Linux Rabbit acquires valid SSH accounts through **brute force**.

10) MULTI-CLASS CLASSIFIER

This step of the pipeline consists in identifying the MITRE ATT&CK tactic most likely described in each tweet message considered normal by the One-Class classifier described above. To accomplish this task, we employed a multi-class machine learning algorithm.

There are multiple ML techniques to perform text classification tasks, from shallow to deep learning algorithms. Shallow algorithms are those which consist of very few layers of composition, including Artificial Neural Network (ANN) with one hidden layer, and that require an expert human to properly extract the features from data that will be learned by the algorithm [45].

On the other hand, deep learning algorithms are those which consist of multiple layers of a composition that aim to avoid the feature extraction process made by humans by automatically learning features from data.

Despite the advantage of automatically learning features from data, deep learn algorithms have well-known limitations

[18]https://scikit-learn.org/stable/index.html

[19]https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html

related to high computational cost in the training phase and more importantly, over-fitting problems when the data set is small. Over-fitting occurs when the machine learning model achieves a good fit on training data but does not generalize on new, unseen data. Nevertheless, we intend to perform tests with deep learning algorithms to be sure if they are inappropriate or not.

To select the best-suited model for our classification problem, which consists in assign one of the 14 MITRE ATT&CK tactics to every analyzed tweet message, we performed benchmark with the following models: Logistic Regression [46], Multinomial Naïve Bayes [47], Linear Support Vector Machine [48], and Random Forest [49]. These models are the most popular and accurate multi-class classification methods in the text classification research domain [50].

For the benchmark, we performed a 10-fold cross-validation using the MITRE ATT&CK corpus using Skitlearn model selection cross_val_score method.[20] The results are shown in Figure 3.

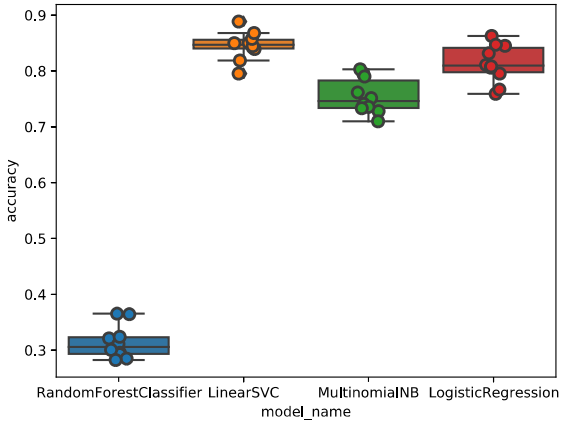

FIGURE 3. ML model benchmark results.

The accuracy results for each model are shown in Table 3.

TABLE 3. Model accuracy comparison benchmark results.

Model Name	Accuracy
LinearSVC	0.845376
LogisticRegression	0.813253
MultinomialNB	0.754975
RandomForest	0.314154

Based on our benchmark results and despite the Random Forest being considered by the literature as a good model for text classification, in our specific domain, it did not produce good results (mean accuracy of 0.31). On the other side, the Linear SVC model presented better results, with a mean accuracy of 0,85 and the smallest variance. Therefore, the Linear SVC model was chosen for our proposed solution.

[20]https://scikit-learn.org/stable/modules/generated/sklearn.model _selection.cross_val_score.html

Linear SVC (Support Vector Classification)[21] is a SVM [51] implementation from Skitlearn.[22]

The machine learning model was trained with the MITRE ATT&CK procedures grouped by the tactic. The procedures consist of text descriptions of how attackers use techniques to achieve their tactics goals. In the table 4, there are some examples of procedures for 3 of 14 tactics of MITRE ATT&CK.

So, the multi-class model consists of 14 classes trained with the MITRE ATT&CK's procedures grouped by the tactic.

TABLE 4. MITRE ATT&CK procedures examples.

Tactic	Procedure
Initial Access	APT18 actors leverage legitimate credentials to log into external remote services.
Initial Access	APT39 has used SQL injection for initial compromise.
Persistence	Lokibotś second stage DLL has set a timer using "timeSetEvent" to schedule its next execution.
Persistence	Grandoreiro can use malicious browser extensions to steal cookies and other user information.
Impact	Avaddon encrypts the victim system using a combination of AES256 and RSA encryption schemes.
Impact	Babuk can stop specific services related to backups.

11) PROCESS IDENTIFIED AND CLASSIFIED THREATS

This step of the pipeline receives the identified threats along with the tweet messages they appear in and the associated MITRE ATT&CK tactic identified by the Multi-class classifier. This data is then enriched and inserted into the Threat Profiling database - refer to the proposed architecture in Figure 1.

The data enrichment consists in aggregating metadata and references to external resources related to the identified cyber threat. In addition to the attributes of the identified threat name and its respective tactic, we aggregate some count attributes coming from tweets the threats appear in with the objective of giving more evidence to threat alerts with higher numbers of mentions, retweets, favorites, followers, and reply.

The list of fields to be inserted into the Threat Profiling database is listed below:

- Date/time: date and time the threat was identified by the solution;
- Threat's name: the name of the identified threat;
- Threat's tactic name: MITRE ATT&CK tactic name identified by the solution;
- Threat's tactic ID: corresponding MITRE ATT&CK tactic's ID;
- Mentions count: the number of times the threat name appeared in tweets;
- Retweets count: the sum of retweets of tweets the threat name appears in;

[21]https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html#sklearn.svm.LinearSVC

[22]https://scikit-learn.org/stable/modules/svm.html

- Favorited count: the sum of favorites received by all tweets the threat name appears in;
- Followers count: the sum of followers of all the profiles which tweeted the content the threat name appears in;
- Reply count: the sum of replies received by all the tweets the threat name appears in;
- References: (URLs) to external content (if exists) from tweets. This can be valuable to threat analysts to have additional information and context about the identified cyber threats;

Once the data is inserted, the next stage of the pipeline, Alert Generation, is triggered.

12) ALERT GENERATION

This is the final step of the pipeline. Its objective is to generate alert messages to cyber threat analysis regarding recently identified cyber threats.

Each alert consists of the threat name and the respective Mitre ATT&CK tactic and has the objective to make analysts aware of the identified threat and its tactics. This way, after further analysis of the threat, defensive actions can be taken to improve the security controls against the threat. For example, if a new threat tactic is to get initial access into the target exploring a vulnerability in a public-facing application, such as Tomcat Web Application, analysts may rapidly access if the vulnerability is present and take mitigation.

One way to make the output of the proposed system more integrated with the defensive team routine is to make the event management system, such as the Security Information Event Management (SIEM), receive alerts and display them to the analysts in a central console.

IV. RESULTS

This chapter presents the results achieved by running our proposed solution, since the data retrieval from the tweets database to the generated alerts. The implementation of the pipeline was automated with scripts developed in Python.

To make it easier to follow each step of the pipeline and its results, we are going to use the diagram in Figure 4, which represents the macro steps of the pipeline, highlighting each step in the following subsections. The full pipeline is described in section III-A.

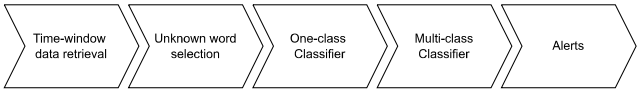

FIGURE 4. Proposed solution pipeline macro diagram.

A. TIME-WINDOW DATA RETRIEVAL

This is the first step of our pipeline. It consists in collecting data from the Tweets Database, described in section III-A2, for a given time range. When in production, the time range will be a sliding time window.

However, for the purposes of this experiment, we considered 30 aleatory 12 hours intervals of time for weekdays from the first semester of 2020. The objective is to simulate the execution of the system in those different time windows and check the results. The time ranges considered and the total tweets collected for each day are shown in Table 5. The total number of tweets collected for the experiment was 90154.

TABLE 5. Tweets time ranges and a respective number of messages.

Time Ranges		
From	To	No. of Tweets
2020-01-06T11:00:00Z	2020-01-06T23:00:00Z	3299
2020-01-09T11:00:00Z	2020-01-09T23:00:00Z	2211
2020-01-13T11:00:00Z	2020-01-13T23:00:00Z	2372
2020-01-21T11:00:00Z	2020-01-21T23:00:00Z	3229
2020-01-24T11:00:00Z	2020-01-24T23:00:00Z	3342
2020-01-28T11:00:00Z	2020-01-28T23:00:00Z	3144
2020-02-04T11:00:00Z	2020-02-04T23:00:00Z	3799
2020-02-10T11:00:00Z	2020-02-10T23:00:00Z	3211
2020-02-12T11:00:00Z	2020-02-12T23:00:00Z	2421
2020-02-19T11:00:00Z	2020-02-19T23:00:00Z	2520
2020-02-24T11:00:00Z	2020-02-24T23:00:00Z	2920
2020-02-28T11:00:00Z	2020-02-28T23:00:00Z	3193
2020-03-03T11:00:00Z	2020-03-03T23:00:00Z	2619
2020-03-05T11:00:00Z	2020-03-05T23:00:00Z	2043
2020-03-12T11:00:00Z	2020-03-12T23:00:00Z	3195
2020-03-17T11:00:00Z	2020-03-17T23:00:00Z	2725
2020-03-30T11:00:00Z	2020-03-30T23:00:00Z	3462
2020-03-31T11:00:00Z	2020-03-31T23:00:00Z	3553
2020-04-01T11:00:00Z	2020-04-01T23:00:00Z	3458
2020-04-07T11:00:00Z	2020-04-07T23:00:00Z	2573
2020-04-09T11:00:00Z	2020-04-09T23:00:00Z	2424
2020-04-15T11:00:00Z	2020-04-15T23:00:00Z	3392
2020-04-22T11:00:00Z	2020-04-22T23:00:00Z	2798
2020-04-29T11:00:00Z	2020-04-29T23:00:00Z	3094
2020-05-01T11:00:00Z	2020-05-01T23:00:00Z	2578
2020-05-05T11:00:00Z	2020-05-05T23:00:00Z	3339
2020-05-08T11:00:00Z	2020-05-08T23:00:00Z	2764
2020-05-11T11:00:00Z	2020-05-11T23:00:00Z	2692
2020-05-12T11:00:00Z	2020-05-12T23:00:00Z	4800
2020-05-20T11:00:00Z	2020-05-20T23:00:00Z	2984
Total		90154

B. UNKNOWN WORD IDENTIFICATION

The step of Unknown word selection, described in section III-A4, consists in identifying words mentioned in tweets that are potential names of cyber threats.

Each set of tweets resulting from the previous step passed through the stage of the Unknown Word Selection step of the pipeline. In the end, this process resulted in 76 unknown words which appeared in 241 tweets. The unknown words are listed in Table 6.

TABLE 6. List of unknown words identified.

76 Unknown Words
'hölzel', 'creds', 'rowhammer', 'cookiethief', 'dacls', 'trickbot', 'emotet', 'lokibot', 'lockbit', 'darkhotel', 'zloader', 'deathransom', 'grbit', 'quidd', 'vollgar', 'pupyrat', 'bisonal', 'kpot', 'draytek', 'volexity', 'uyghur', 'sanix', 'doppelpaymer', 'kaiji', '16shop', 'thunderspy', 'dridex', 'citrixadc', 'codinglikeitsthe90s', 'phising', 'apt41', 'turla', 'necurs', 'unsandboxed', 'apt36', 'lazarus-group', 'robbinhood', 'metamorfo', 'drbcontrol', 'discsans', 'dreambot', 'astaroth', 'wannacry', 'kwampirs', 'houseparty', 'revil', 'troldesh', 'oceanlotus', 'dast', 'skywrapper', 'hancitor', 'ragnarlocker', 'nemty', 'scammers', 'phishers', 'sodinokibi', 'repurposing', 'prepper', 'ftcode', 's4x20', 'iconics', 'genesis64', 'threattraq', 'emcor', 'azorult', 'anime', 'locky', 'wolfrat', 'icedid', 'bokbot', 'ursnif', 'loda', 'kbot', 'iloveyou', 'xhelper', 'cacheout'

TABLE 7. List of unknown words which are cyber threats.

55 Unknown words which are cyber threats
rowhammer, cookiethief, dacls, trickbot, emotet, lokibot, lockbit, darkhotel, zloader, deathransom, vollgar, pupyrat, bisonal, kpot, draytek, sanix, doppelpaymer, kaiji, 16shop, thunderspy, dridex, apt41, turla, necurs, apt36, lazarusgroup, robbinhood, metamorfo, drbcontrol, dreambot, astaroth, wannacry, kwampirs, revil, troldesh, oceanlotus, dast, skywrapper, hancitor, ragnarlocker, nemty, sodinokibi, ftcode, s4x20, azorult, locky, wolfrat, icedid, bokbot, ursnif, loda, kbot, iloveyou, xhelper, cacheout

TABLE 8. List of unknown words which are not cyber threats.

21 Unknown words which are not cyber threats
hölzel, creds, grbit, quidd, volexity, uyghur, citrixadc, codinglikeitsthe90s, phising, unsandboxed, discsans, houseparty, scammers, phishers, repurposing, prepper, iconics, genesis64, threattraq, emcor, anime

An experienced cyber security specialist verified the resulting list of unknown words to verify which of those are cyber threats and which are not. The verification showed that, from 76 unknown words selected by the 'Unkown Word Selection' step, 56 are threats and 20 are not as shown in Tables 7 and 8, respectively.

1) DATA LABELLING

An additional procedure was taken in this step of the pipeline to prepare the data to be validated by the next two steps of the pipeline, which consist in classifying data using two machine learning models.

The additional step consisted in labeling each of 241 Twitter messages returned by 'Unknown Word Selection'. The labeling was performed by a cyber security specialist that read carefully each tweet message and associated a label according to the following rules:

1) If the tweet mentions or describes, even partially, one of the 14 MITRE ATT&CK tactics, a label with the corresponding tactic is associated with the Tweet message;
2) If the tweet message does not describe any of MITRE ATT&CK tactics, a label 'NOLABEL' was associated.

To make the labeling job simpler and more organized, we use a tool called Doccano.[23] Doccano is an open-source text annotation tool to create labeled data for sentiment analysis, named entity recognition, text summarization, and so on.

In Figure 5, there is an example of a Tweet message being labeled using Doccano tool. The message says that Cookiethief, the discovered threat, exfiltrates browser and Facebook app cookies to a malicious server. Thus the 'Exfiltration' tactic was associated with the message.

By the end of the labeling process, from 241 tweets, 92 tweets were associated with a MITRE ATT&CK tactic and 149 were not (NOLABEL), as seen in Table 9.

From the total of 76 unknown words, 42 are present in the 92 tweets associated with at least one Mitre ATT&CK tactic (Table 10). The other 34 unknown words had no association with any tactic (Table 11).

[23]https://doccano.github.io/doccano/

TABLE 9. Tweets labelling results classified by MITRE ATT&CK Tactics.

Tactic	Count
Initial Access	25
Execution	15
Persistence	6
Defense Evasion	13
Credential Access	8
Discovery	3
Lateral Movement	4
Collection	5
Command and Control	9
Exfiltration	2
Impact	2
NOLABEL	149
TOTAL	**241**

TABLE 10. Unknwon words with at least one tweet associated with a tactic.

42 unknown words with at least one tweet associated with a tactic
hölzel, creds, cookiethief, dacls, trickbot, emotet, lokibot, lockbit, darkhotel, zloader, deathransom, grbit, quidd, pupyrat, bisonal, kpot, draytek, sanix, doppelpaymer, kaiji, 16shop, dridex, apt36, lazarusgroup, robbinhood, thunderspy, astaroth, oceanlotus, hancitor, vollgar, ftcode, s4x20, iconics, genesis64, emcor, azorult, wolfrat, ursnif, loda, xhelper, cacheout , dreambot

TABLE 11. Unknwon words without tweets associated with tactics.

34 unknown words without tweets associated with tactics
rowhammer, volexity, uyghur, citrixadc, codinglikeitsthe90s, phising, apt41, turla, necurs, unsandboxed, metamorfo, drbcontrol, discsans, wannacry, kwampirs, houseparty, revil, troldesh, dast, skywrapper, ragnarlocker, nemty, scammers, phishers, sodinokibi, repurposing, prepper, threattraq, anime, locky, icedid, bokbot, kbot, iloveyou

C. ONE-CLASS CLASSIFICATION

This step of the pipeline consists in using a machine learning model to select tweets whose content is close to the description of malicious actions while discarding the ones that are not. To do this, we implemented a One-class classifier, as detailed in III-A8. The model decides whether a tweet message is normalsimilar or abnormaldifferent from the training set. The normal messages are selected and forwarded to the next step of the pipeline and the abnormal messages are discarded. As a reminder, the training set for the One-class model consists of all the MITRE ATT&CK techniques' threat procedures examples.

The One-class classifier was implemented using OneClassSVM algorithm from SkitLearn and was trained with all the procedures of MITRE ATT&CK, regardless of the tactic or technique to which it belongs, as described in section III-A8. The training data consists of a TF-IDF feature matrix created from the MITRE ATT&CK procedures corpus using the implementation of the class TfidfVectorizer from Skitlearn, as described in section III-A7.

The TfidfVectorizer class has a parameter named min_df which is used to inform the algorithm of a threshold to ignore terms when building the vocabulary that has a document

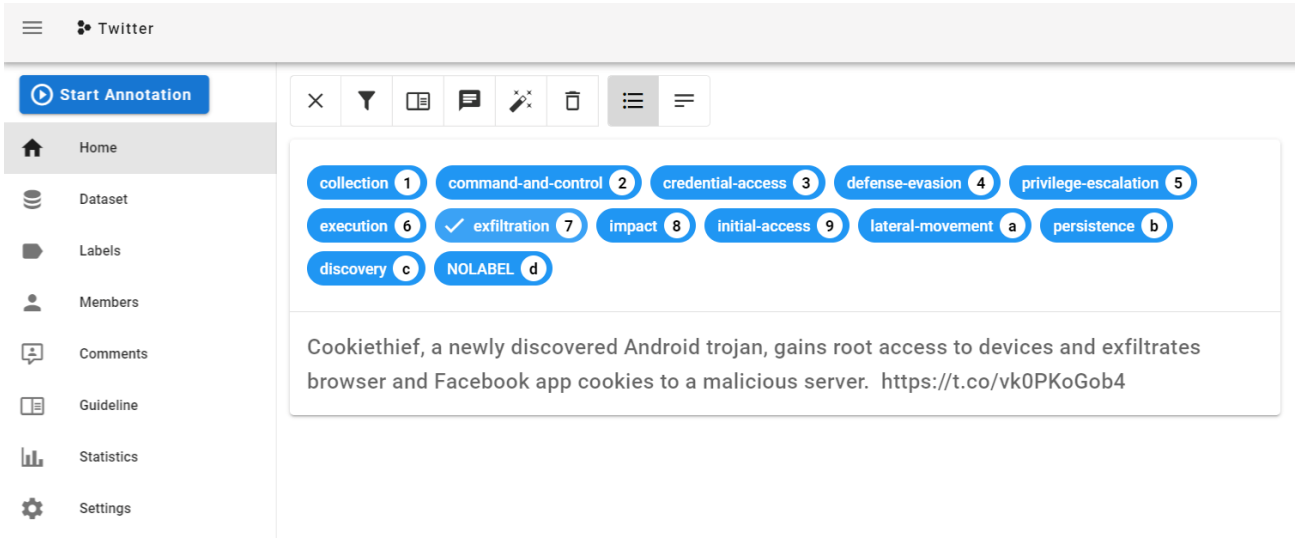

FIGURE 5. Labelling Tweets using Doccano - sample data.

TABLE 12. One-Class results for min_df variations from 1 to 4 for tweets classification.

	Normal			Abnormal			Metrics		
	Selected	TP	FP	Discarded	TN	FN	Recall	Precision	Accuracy
min_df=1	78	59	19	163	130	33	64.13%	75.64%	78.42%
min_df=2	40	34	6	201	143	58	36.95%	85.00%	73.44%
min_df=3	29	25	4	212	145	67	27.17%	86.20%	70.54%
min_df=4	28	23	5	213	144	69	25.00%	82.14%	69.29%

frequency strictly lower than it. The default parameter for min_df is 1, which means ignoring terms that appear in less than 1 document.

However, in this experiment, we increased this number to make the algorithm ignore more infrequent terms while building the One-class training set to check its influence on classification performance.

Before we pass the tweet messages to the One-class classifier, we must transform the tweets into a document-term matrix using the vocabulary and the term frequency (TF) produced in the One-class training phase. To do this, we used the method transform from TfidfVectorizer.

The results for the One-Class classification of 241 labeled tweets for 'min_df' ranging from 1 to 4 are shown in 12.

Our experiments showed that the higher the value of min_df, the lower the number of selected tweets in this stage of the pipeline. For min_df of 1, there were selected 78, for min_df of 2, 40 tweets, for min_df of 3, 29 tweets, and for min_df of 4, 28 tweets.

The best accuracy was 78.42% when the 'min_df' value is set to 1. On the other hand, the precision, which means the proportion of positive identifications actually correct, is better for 'min_df' 2 and 3, with values of 85% and 86.20% respectively.

D. MULTI-CLASS CLASSIFICATION

This step of the pipeline consists in identifying the MITRE ATT&CK tactic most likely described in each tweet message considered normal by the One-Class classifier run in the previous step of the pipeline.

As described in Section III-A10 the multi-class model consists of 14 classes trained with the MITRE ATT&CK's procedures grouped by the tactic. Similarly to what was performed for the One-class classifier, the training data for the multi-class consists of a TF-IDF feature matrix created from the MITRE ATT&CK procedures. The difference is that for the multi-class the procedure corpus was divided into 14 tactics, one for each class.

To implement the classifier, we used the Linear SVC (Support Vector Classification) implementation from Skitlearn.[24]

To compare the results, we executed the Multi-Class step for each One-Class result. As a reminder, the One-Class results were processed using values from 1 to 4 for the min_df parameter.

Multi-class results are presented in Tables 13, 14, 15 and 16.

We performed experiments with higher values for min_df until 10 to check the influence of this parameter in the F1-score performance for results for both One-class and Multi-class classifiers. The results are shown in Figure 6.

Given the results comparing different values for min_df parameter, the higher the min_df parameter until the limit of 3, we noticed that:

[24]https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_score.html

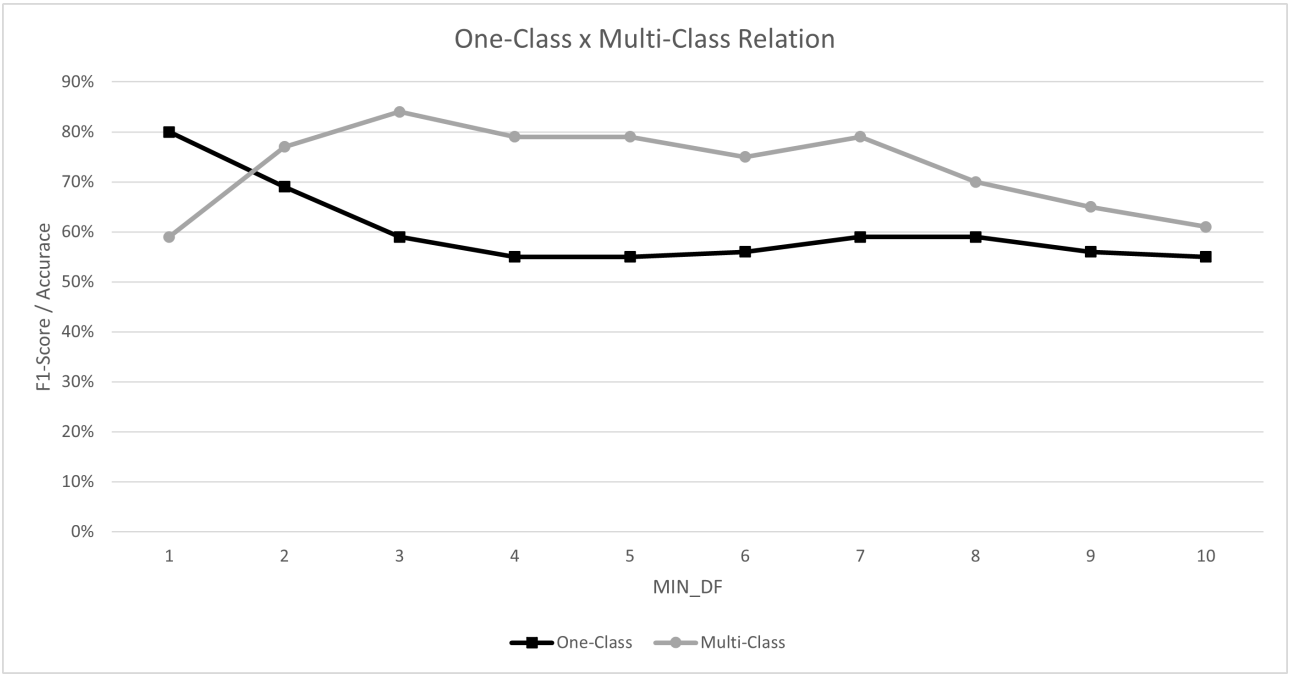

FIGURE 6. One-Class and Multi-Class relation in terms of min_df parameter.

TABLE 13. Multi-class results for One-class min_df=1.

Tactic	Precision	Recall	F1-Score	Suport
NOLABEL	0.00	0.00	0.00	19
Collection	1.00	1.00	1.00	4
Command-and-control	0.55	0.86	0.67	7
Credential Access	1.00	1.00	1.00	8
Defense Evasion	0.50	1.00	0.67	9
Discovery	0.23	1.00	0.38	3
Execution	0.20	0.33	0.25	3
Exfiltration	0.00	0.00	0.00	1
Impact	0.33	1.00	0.50	1
Initial Access	0.90	0.60	0.72	15
Lateral Movement	1.00	0.40	0.57	5
Persistence	1.00	0.40	0.57	5
Micro AVG	0.60	0.59	0.59	78
Macro AVG	0.56	0.68	0.56	78
Weighted AVG	0.56	0.59	0.53	78

- the lower is the number of tweet messages resulting from one-class classifier and automatically the lower is the number of input tweets for the multi-class classifier;
- the lower is the accuracy of one-class classifier;
- the higher is the false negative of one-class classifier;
- the higher is the precision of the one-class classifier;
- the higher is the F1-Score of the multi-class classifier.

From this, we can conclude that the min_df parameter is turning the one-class classifier more restrictive, making it select tweet messages closest to the description of a malicious behavior described in MITRE ATT&CK procedures. This is reflected in better F1-Scores in multi-class classifier.

The downside of having a more restrictive one-class is the increase of false negatives for the one-class. This means that we mistakenly discard tweets that should not be discarded.

TABLE 14. Multi-class results for One-class min_df=2.

Tactic	Precision	Recall	F1-Score	Suport
NOLABEL	0.00	0.00	0.00	6
Collection	1.00	1.00	1.00	3
Command-and-control	0.71	0.83	0.77	6
Credential Access	1.00	1.00	1.00	6
Defense Evasion	0.56	1.00	0.71	5
Discovery	0.60	1.00	0.75	3
Impact	1.00	1.00	1.00	1
Initial Access	1.00	0.80	0.89	5
Lateral Movement	1.00	1.00	1.00	2
Persistence	1.00	0.33	0.50	3
Micro AVG	0.79	0.75	0.77	40
Macro AVG	0.79	0.80	0.76	40
Weighted AVG	0.72	0.75	0.71	40

TABLE 15. Multi-class results for One-class min_df=3.

Tactic	Precision	Recall	F1-Score	Suport
NOLABEL	0.00	0.00	0.00	4
Command-and-control	0.75	0.86	0.80	7
Credential Access	1.00	1.00	1.00	8
Defense Evasion	0.67	1.00	0.80	2
Discovery	0.67	1.00	0.80	2
Impact	1.00	1.00	1.00	1
Initial Access	1.00	1.00	1.00	1
Lateral Movement	0.67	1.00	0.80	2
Persistence	1.00	0.50	0.67	2
Micro AVG	0.82	0.79	0.81	29
Macro AVG	0.75	0.82	0.76	29
Weighted AVG	0.73	0.79	0.75	29

The balance value for the min_df value can be interpreted this way: if your system is more important to identify new threats than the accuracy of the respective tactics, a value of 1 is suitable for min_df. If the accuracy of identifying the

TABLE 16. Multi-class results for One-class min_df=4.

Tactic	Precision	Recall	F1-Score	Suport
NOLABEL	0.00	0.00	0.00	5
Command-and-control	0.62	1.00	0.77	5
Credential Access	1.00	1.00	1.00	8
Defense Evasion	0.67	1.00	0.80	2
Discovery	0.67	1.00	0.80	2
Initial Access	1.00	1.00	1.00	2
Lateral Movement	0.67	1.00	0.80	2
Persistence	1.00	0.50	0.67	2
Micro AVG			0.79	28
Macro AVG	0.70	0.81	0.73	28
Weighted AVG	0.68	0.79	0.71	28

TABLE 17. Alert list.

ID	Unknown Word	Is a threat?	Tactic classified	Tactic correctly identified?
1	16shop	True	initial-access	True
2	16shop	True	initial-access	True
3	bisonal	True	defense-evasion	True
4	creds	False	command-and-control	True
5	dacls	True	discovery	False
6	darkhotel	True	discovery	False
7	darkhotel	True	defense-evasion	False
8	deathransom	True	impact	True
9	doppelpaymer	True	collection	True
10	doppelpaymer	True	collection	True
11	draytek	True	command-and-control	True
12	emotet	True	command-and-control	True
13	emotet	True	discovery	True
14	emotet	True	credential-access	True
15	emotet	True	defense-evasion	True
16	emotet	True	lateral-movement	True
17	emotet	True	defense-evasion	False
18	emotet	True	collection	True
19	grbit	False	defense-evasion	True
20	holzel	False	discovery	True
21	kaiji	True	credential-access	True
22	kpot	True	collection	False
23	lockbit	True	discovery	True
24	lokibot	True	defense-evasion	True
25	pupyrat	True	command-and-control	True
26	quidd	False	credential-access	True
27	sanix	True	credential-access	True
28	sanix	True	credential-access	True
29	trickbot	True	persistence	True
30	trickbot	True	execution	False
31	trickbot	True	defense-evasion	True
32	trickbot	True	lateral-movement	True
33	trickbot	True	initial-access	True
34	trickbot	True	credential-access	True
35	uyghur	False	privilege-escalation	False
36	uyghur	False	command-and-control	False
37	volexity	True	command-and-control	False
38	vollgar	True	defense-evasion	False
39	zloader	True	command-and-control	True
40	zloader	True	command-and-control	True

respective tactics for the identified threats is more important than the number of threats, a min_df of 3 is most appropriate.

In our experiment, as seen in Figure 6, a good trade-off has been achieved using a min_df = 2. With this, we had 69% for One-Class and 77% for the Multi-Class.

E. ALERTS

This is the last step of the pipeline. It consists of generating alerts to cyber threat analysts regarding identified threats and corresponding identified tactic.

TABLE 18. Experiment summary.

Result	Value	%
Processed tweets	90154	100.00%
Tweets with uncommon words	241	0.27%
Tweets resulted from One-class classifier	40	0.04%
Total number of alerts	40	100.00%
Alerts with real threats	34	85.00%
Alerts with false threats	6	15.00%
Unique uncommon words identified	23	100.00%
Unique real threats identified	18	78.26%
Tactics associated to tweets	40	100.00%
Tactics correctly associated to tweets	30	75.00%

TABLE 19. Execution summary.

Result	Value	%
Processed tweets	204718	100.00%
Tweets with uncommon words	1290	0.63%
Tweets resulted from One-class classifier	292	0.14%
Total number of alerts	292	100.00%
Alerts with real threats	181	61.99%
Alerts with false threats	111	38.01%
Unique uncommon words identified	74	100.00%
Unique real threats identified	40	54.05%
Tactics associated to tweets	292	100.00%
Tactics correctly associated to tweets	173	59.00%

The input for the alert generation procedure is the output of the multi-class classification. As described in the previous section, we executed the multi-class step using different values for the variable min_df. However, the results we are going to present in this section are for the min_df of 2, which resulted in 40 classified tweets.

The alert procedure generated a total of 40 alerts detailed in Table 17. From those, 34 were alerts for real threats and a total of 30 tactics were correctly identified.

The Table 18 summarizes the results of the entire experiment:

V. DISCUSSION

To evaluate the proposed solution, we implemented the pipeline to automatically collect, process, and generate alerts online for the Threat Intelligence Team of a big financial institution in Brazil.

The solution runs from May 26, 2021, to August 04, 2021. All the pipeline was automated using Python, from the data collection to alert generation, following the solution characteristics described in this research.

The summary for the execution is presented in Table 19.

The number of analyzed tweets (204,718) compared to the number of alerts (212) gives an idea of how much work has been saved from analysts in the job of hunting for emerging cyber threats by our solution. On the other side, the number of false positives was high (38.01%) and is the subject of future research. False alerts brought analysts unusual names that were not necessarily names of threats. This improvement must be addressed both in Unknown Word Selection and One-class steps of the proposed pipeline as they select the tweets that go to the final classification phase - the Multi-class classifier. However, improvement is challenging given some uncommon words can appear in tweets that mention

TABLE 20. Identified threats online.

Threat	Tactic	Tweet example	Analysis
petitpotam	Privilege Scalation	ADCSPwn - a tool to **escalate privileges** in an active directory network by coercing authenticate from machine accounts (**Petitpotam**) and relaying to the certificate service.	Threat name and tactic correctly identified.
translogic	Credential Access	Multiple zero-day flaws were found in the Swisslog Healthcare's Translogic PTS system, a transport system used in 3,000+ hospitals. Dubbed **'PwnedPiper'**, the bugs could let attackers disrupt delivery of lab samples or **steal hospital employee credentials**: https://t.co/l79v7LAs6A	The name of the threat was incorrectly identified. The right name is PwnedPiper. The tactic was correctly identified.
solarmarker	Collection	**#Solarmarker** is an **information stealer** that we've seen increase in activity since September. Here's why this #malware is after your data, and what you can do to protect your network https://t.co/U8vyS698EJ https://t.co/pT7GXZcqwR	Threat name and tactic correctly identified.
ta551	Defense Evasion	@malware_traffic It is looking like this is defiantly a **TA551** Specific **"Obfuscation Format"** Dropping different end malware.	Threat name and tactic correctly identified. Obfuscation are usually used to bypass security controls.
revil	Defense Evasion	@wackygiraffe91 That's how universal decryptors work, though. It's in the article. **REvil** has 4 sets of encryption keys that the REvil coders can use to **bypass unruly affiliates**.	Threat name correctly identified. The Defense Evation tactic came from the "bypass" in the Twitter but it is related to bypass affiliates and not security controls.
anchordns	Command and Control	Kryptos Logic researchers write about changes in the **AnchorDNS** backdoor used by TrickBot and Bazar malware campaigns. They observed that the **C2 communications** protocol of AnchorDNS has changed and the use of another Anchor component called AnchorAdjuster. https://t.co/whSJJxiLg3 https://t.co/VwtjMRxNGr	Threat name and tactic correctly identified.
nobelium	Credential Access	Microsoft says **Nobelium**, the group behind the SolarWinds hack, aka the SVR, has breached:-three new victims (**through brute-force attacks**) -a MSFT's tech support agent's device (via malware) https://t.co/vUAypYKLqB https://t.co/bFmX3TwQKX	Threat name and tactic correctly identified.
babuk	Impact	We've got our hands on **Babuk**'s Builder, the ransomware group that **ransomed The DC Police Department** and CD Projekt Red. This creates both the payload and the decryptor. Download it here: vxug.fakedoma{[}.{]}in/tmp/ * Link modified to conform with Twitters ban on our domains https://t.co/bFt2wh8Z1W	Threat name and tactic correctly identified.
emotet	Defense Evasion	The **Emotet** malware, as well as some of it's modules, have been completely redesigned, improving **anti-malware evasion**. The botnets also implement hashbusting, which ensures the malware's file hash is different on each infected system. 2/4	Threat name and tactic correctly identified.
trickbot	Command and Control	More conversations from the recruiting process. Apparently, the **Trickbot** gang had their own **private communications server**. I'll presume based on the court docs that they were running their own internal Jabber/XMPP service. https://t.co/MyHVHQcpKD	Threat name and tactic correctly identified.
mosaicloader	Command and Control	A never-before-documented #Windows #malware strain dubbed **#MosaicLoader** is spreading worldwide, acting as a full-service malware-delivery platform that's being used to infect victims with #RATs, @Facebook cookie stealers, etc. #cybersecurity @Bitdefender https://t.co/u3uYV9Iz3j	Threat name correctly identified. The Command and Control tactic is not explicitly present on Twitter.

malicious behavior but the word itself does not name a cyber threat. For example, one of the selected tweets includes the word 'MonPass' in the text *'The attackers used their access to distribute a backdoored version of the MonPass client app'*. Despite not naming a threat, but the name of a product, the tweet in which it appears led the One-class algorithm to select it given the similarity to texts describing malicious behavior.

A sample of the generated alarms is shown in Table 20. Along with each identified cyber threat, we included an analysis of the result by a cyber security specialist assessing whether the threat name and tactic have been correctly identified.

Three alerts were especially important to the cyber security team as they made the team aware of emerging cyber threats. The alerts and their importance are detailed below.

A. PETITPOTAM

PetitPotam was the name given to a cyber threat discovered by the French researcher Gilles Lionel (nicknamed PetitPotam), able to exploit a vulnerability in Windows operating system to force remote Windows servers to authenticate with an attacker and share NTLM authentication details or authentication certificates in an attack known as NTLM Relay Attack. Using this technique, an unauthenticated attacker

could completely take over a Windows domain with the Active Directory Certificate Service (AD CS) running — including domain controllers.

The first alert for the term PetitPotam, was issued by our system on July 24, 2021. This day, the term 'PetitPotam' was mentioned 63 times by different monitored profiles along with threat terms such as 'attack' and 'exploit'. During the next days, the term was alerted again as the term continued to be mentioned by the monitored Twitter profiles.

The alert made the cyber threat intelligence team aware of the threat and its tactic. After further investigations, the team was able to find out that Microsoft released the day before (July 23, 2021) a security advisory with instructions to mitigate PetitPotam vulnerability.[25] The official patch for the vulnerability exploited by PetitPotam would be available on August 10, 2021 (CVE-2021-36942[26]).

So, in summary, since the emerging of the term PetitPotam on Twitter, the cyber security team was made aware of the threat and its intents and started to work on mitigation 17 days before the official patch was made available by Microsoft.

B. MOSAICLOADER

MosaicLoader was the name given to malware being used to infect victims with remote-access trojans (RATs), Facebook cookie stealers, and other threats. According to the BitDefender, the company that found the threat, the malware is spreading indiscriminately worldwide through paid ads in search results, targeting people looking for pirated software and games. It masquerades as a cracked software installer, but in reality, it's a downloader that can deliver any payload to an infected system.

The alert for MosaicLoader was issued by the system on July 21, 2021, one day after BitDeffender published the report about the new threat.[27] The term was mentioned 12 times by the monitored profiles and was associated with threat words: spyware, infected, malware, rat, and rce.

C. EMOTET

Emotet is a malware originally developed as a banking trojan. First identified by security researchers in 2014[28] it has been evolving ever since to improve its capabilities like spamming and the ability to work as a malware delivery service, including other banking malware.

Although in this case, the system did not identify a new threat, the alert remains important as it signals the return of the threat after months of inactivity and makes analysts aware of new malware capabilities related to defense evasion tactics.

Following the other tweets from the same profile (@MalwareTechBlog), it was possible to understand the defense evasion improvements. The tweet says *'New Emotet code utilizes a state machine to obfuscate control flow. Branches are flattened into nested loops, allowing code blocks to be placed in arbitrary order, with flow controlled by a randomized state value. This allows for easy code mutation and possibly polymorphism.'*.

This means that the defense team must improve and test technical controls to make sure they could catch the new variant before a real compromise. This includes making simulations with the new variant sample on controlled machines prepared with the company's defense technologies.

These three sample alerts exemplify how the system can make, in a timely and automated manner, a cyber security team aware of emerging threats based on Twitter mining. The idea is exactly having the system identify and minimally profile cyber threats as early as possible and dedicate specialized people to refine the threat profile to apply countermeasures instead of reading hundreds or thousands of tweets every day to pinch something important.

VI. CONCLUSION

Given the dynamism of the cyber security field, with new vulnerabilities and threats appearing at any time, keeping up to date on them is a challenging but important task for analysts. Even following the best practices and applying the best controls, a new threat may bring an unusual way to subvert the defenses requiring a quick response. This way, timely information about emerging cyber threats becomes paramount to a complete cybersecurity system.

This research proposes an automated cyber threat identification and profiling based on the natural language processing of Twitter messages. The objective is exactly to cooperate with the hard work of following the rich source of information that is Twitter to extract valuable information about emerging threats in a timely manner.

This work differentiates itself from others by going a step beyond identifying the threat. It seeks to identify the goals of the threat by mapping the text from tweets to the procedures conducted by real threats described in MITRE ATT&CK knowledge base. Taking advantage of this evolving and collaborative knowledge base to train machine learning algorithms is a way to leverage the efforts of cyber security community to automatically profile identified cyber threats in terms of their intents.

To put in test our approach, in addition to the research experiment, we implemented the proposed pipeline and run it for 70 days generating online alerts for the Threat Intelligence Team of a big financial institution in Brazil. During this period, at least three threats made the team take preventive actions, such as the PetitPotam case, described in section V. Our system alerted the team making them aware of PetitPotam 17 days before the official patch was published by Microsoft. Within this period, the defense team was able to implement mitigations avoiding potential exploits and, consequently, incidents.

[25] https://msrc.microsoft.com/update-guide/vulnerability/ADV210003

[26] https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-36942

[27] https://www.bitdefender.com/blog/labs/debugging-mosaicloader-one-step-at-a-time

[28] https://www.malwarebytes.com/emotet

Our experiments showed that the profiling stage reached an F1 score of 77% in correctly profiling discovered threats among 14 different tactics and the percentage of false alerts of 15%. In future work, we consider it important to advance in tweets selection stages (Unknown Words and One-class), to improve the false positives rate and in the profiling stage, to reach higher accuracy in determining the technique associated with the identified threat. We are working on this way by experimenting with a different NLP approach using the part of speech (POS) algorithm implementation from Spacy[29] Python library. The object is to identify the root verb, the subject, and the object of the phrases to select tweets where the action described (the root verb) is referencing the unknown word (the subject).

APPENDIX A
CYBER THREAT DICTIONARY

See Table 21.

TABLE 21. List of terms.

Cyber Threat Dictionary
0day, actor, attack, bloatware, bot, botnet, brute force, bypass, c2, campaign, crack, cryptojacking, cryptominer, cyberattack, cybercriminal, cyberespionage, cybergang, cyberthreat, data stealer, ddos, dos, downloader, drive by, exfiltrate, exploit, fileless, gang, group, hack, infect, inject, injects, keylogger, malicious, malverts, malware, mitm, payload, pharming, phish, phishing, preinstalled, ransomware, rat, rce, re infecting, redirect, rootkit, scammer, skimming, sniff, spamming, spoof, spyware, state sponsored, threat, trojan, trojanizedunauthorized, virus, watering hole, weaponize, weaponizedworm, zombie, escalate, manipulate, manipulating, abuse, backdoor, steal, capture, collect, dump

APPENDIX B
NORMALIZED TERMS

List of normalized terms.

Unnormalized term	Normalized term
two factor	2fa
two-factor	2fa
two factor authentication	2fa
two-factor-authentication	2fa
malspam	email phishing
smiphing	sms phishing
spearphishing	phishing
spear phishing	phishing
spear-phishing	phishing
spear phishing	phishing
phishing-as-a-service	phishing as a service
brute-force	brute force
bruteforce	brute force
brute-forces	brute force
bruteforces	brute force
maldoc	malicious document
data-wiping	data wiping
datawiping	data wiping
disk-wiping	disk wiping
diskwiping	disk wiping
card-skimming	card skimming
cardskimming	card skimming
c&c	c2
c&c	c2
command and control	c2
command-and-control	c2
distributed denial of service	ddos
distributed-denial-of-service	ddos
denial of service	dos
denial-of-service	dos
denialofservice	dos
0-day	0day
0-days	0day
0day	0day
0days	0day
zero-day	0day
zero-days	0day
zeroday	0day
zerodays	0day
re-infecting	re infecting
text.replace(ransomware-as-a-service	ransomware as a service
indicator of compromise	ioc
indicators of compromise	ioc
indicator-of-compromise	ioc
indicators-of-compromise	ioc
information stealer	data steal
information-stealing	data steal
data-stealing	data steal
infostealer	data steal
data harvesting	data steal
credential-stealing	data steal
data exfiltrating	data exfiltrating
user access control	uac
user-access-control	uac
crypto-mining	cryptomining
crypto-jacking	cryptojacking
crypto mining	cryptomining
crypto jacking	cryptojacking
man-in-the-middle	mitm
man in the middle	mitm
remote desktop	rdp
cyber-attacks	cyberattack
cyber attacks	cyberattack
cyber-attack	cyberattack
cyber attack	cyberattack
cyber-criminal	cybercriminal
cyber criminal	cybercriminal
cyber-criminals	cybercriminal
cybercriminals	cybercriminal
cyber-espionage	cyberespionage
cyber espionage	cyberespionage
cyber-gang	cybergang
cyber gang	cybergang

[29] https://spacy.io/

cyber-gangs	cybergang
cyber gangs	cybergang
cyber-threat	cyberthreat
cyber threat	cyberthreat
cyber-threats	cyberthreat
cyber threats	cyberthreat
e-mail	email
password-spray	password spray
passwordspray	password spray
code-injection	code injection
codeinjection	code injection
sim-swapping	sim swapping
sim-swap	sim swap
remote code execution	rce
remote-code-execution	rce
remote access trojan	rat
remote-access-trojan	rat
state-sponsored	state sponsored
statesponsored	state sponsored
drive-by	drive by
watering-hole	watering hole
water-holing	watering hole
watering-holing	watering hole
file-less	fileless
file less	fileless
key-logger	keylogger
key logger	keylogger
keylogging	keylogger
key-logging	keylogger
record keystrok	keylogger
credential logger	keylogger
credential logging	keylogger
capture keystroke	keylogger
capture keystrokes	keylogger
captures keystrokes	keylogger
log keystroke	keylogger
log keystrokes	keylogger
root-kit	rootkit
root kit	rootkit
skimming-as-a-service	skimming as a service
bot-nets	botnet
botnets	botnet
bot nets	botnet
bot net	botnet
pre-installed	preinstalled
pre installed	preinstalled
by pass	bypass
by-pass	bypass
back-door	backdoor
back door	backdoor
backdoors	backdoor

APPENDIX C
LIST OF TERMS

The list of terms is shown in Table 22.

TABLE 22. List of terms.

Cyber Related Words Dictionary
2fa, admin, admins, adware, air-gapped, airgapped, analytics, anti-phishing, antivirus, apis, app-based, apps, appsec, apts, asian, auth, automate, automation, backend, backends, blacklist, blockchain, blog, blueteam, bluetooth, botnet-associated, bots, bruteforce, bug, built-in, cert, certifiate, ciso, code-signing, config, contactless, cookie, countermeasure, created/operated, cred, credential, credentials, cross-site, crypto, cryptocurrency, cryptographic, cryptoining, cryptomining, customer-facing, customizable, cwbrief, cw-podcast, cyber, cyberattackers, cyberattacks, cybercrime, cybercrime-as-a-service, cybermonday, cybersecurity, cyberspace, cyberthreatintelligence, cyberthreats, cyberwar, darkweb, database, databreach, datasecurity, dataset, dbir, ddos-for-hire, decrypt, decrypter, decryptor, defcon, defense, defenses, demo, devops, dfir, dmarc, dns, docusign, download, downloads, e-book, e-mail, ebook, email, emails, encrypt, encrypted, encrypting, end-to-end, endpoint, exec, executables, factorytalk, firefox, firewall, firmware, follow-up, fuzzing, gdpr, geolocation, ghidra, gigabyte, hackathon, hacktivist, hardcoded, hash, hashes, hdds, honeypot, honeypots, hostname, hostnames, hotfix, html, http, https, hunter, hxxp, icloud, imap, in-depth, in-game, inbox, inboxes, incidentresponse, informationsecurity, infosec, insiderthreat, insiderthreats, installers, installment, integrations, intel, internet, ioc, iocs, iot, iot-security, iphone, iphones, jailbreak, javascript, key, keystroke, laptop, linux, logins, long-running, machinelearning, macos, magento, malspam, marketplace, marketplaces, metasploit, misconfiguration, mitigation, mitigations, mobilesecurity, modeling, monero, monetize, mozilla, multi-factor, nas/nvr, netcat, next-stage, northkorea, o365, office365, online, onlinedanger, open-source, open-sources, operationalizing, osint, outages, p2omiami, pastebin, patchtuesday, pcap, peer-to-peer, pen-testing, pentest, pentesting, php, plug-in, plugin, plugins, podcast, point-of-sale, powershell, pre-installed, productreviews, proof-of-concept, proxy, psexec, raspberrypi-powered, real-time, red-team, redirects, redteam, repo, rfid, s/mime, sandbox, sdks, secret-key, self-isolated, sha-1, sha1, sha256, sha512, siem, smartphone, smartphones, smbv3, socialengineering, software, spam, spoofing, spreadsheet, ssl, sso, startup, stormcast, struct, subnet, supercomputer, sysmon, telco, telnet, third-party, threatdetection, threathunt, threathunting, threatintel, threatintelligence, timeline, training, trainings, ttps, tweet, tweets, uac, unencrypted, unexploitable, unhackable, unpatchable, unpatched, upload, url, urls, username, usernames, vodaphone, voip, vpn, vpns, vuln, vulns, weaponized, webcam, webcast, webhooks, webinar, weblogic, webpage, website, websites, whitepaper, wifi, wiki, womenincyber, wordpress, workaround, workarounds, wormable, xbox, yara, youtube, zero-day, zero-day, zerodays

REFERENCES

[1] B. D. Le, G. Wang, M. Nasim, and A. Babar, "Gathering cyber threat intelligence from Twitter using novelty classification," 2019, *arXiv:1907.01755*.

[2] *Definition: Threat Intelligence*, Gartner Research, Stamford, CO, USA, 2013.

[3] R. D. Steele, "Open source intelligence: What is it? why is it important to the military," *Journal*, vol. 17, no. 1, pp. 35–41, 1996.

[4] C. Sabottke, O. Suciu, and T. Dumitras, "Vulnerability disclosure in the age of social media: Exploiting Twitter for predicting real-world exploits," in *Proc. 24th USENIX Secur. Symp. (USENIX Secur.)*, 2015, pp. 1041–1056.

[5] A. Sapienza, A. Bessi, S. Damodaran, P. Shakarian, K. Lerman, and E. Ferrara, "Early warnings of cyber threats in online discussions," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2017, pp. 667–674.

[6] E. Nunes, A. Diab, A. Gunn, E. Marin, V. Mishra, V. Paliath, J. Robertson, J. Shakarian, A. Thart, and P. Shakarian, "Darknet and deepnet mining for proactive cybersecurity threat intelligence," in *Proc. IEEE Conf. Intell. Secur. Informat. (ISI)*, Sep. 2016, pp. 7–12.

[7] S. Mittal, P. K. Das, V. Mulwad, A. Joshi, and T. Finin, "CyberTwitter: Using Twitter to generate alerts for cybersecurity threats and vulnerabilities," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2016, pp. 860–867.

[8] A. Attarwala, S. Dimitrov, and A. Obeidi, "How efficient is Twitter: Predicting 2012 U.S. presidential elections using support vector machine via Twitter and comparing against Iowa electronic markets," in *Proc. Intell. Syst. Conf. (IntelliSys)*, Sep. 2017, pp. 646–652.

[9] N. Dionísio, F. Alves, P. M. Ferreira, and A. Bessani, "Towards end-to-end cyberthreat detection from Twitter using multi-task learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–8.

[10] O. Oh, M. Agrawal, and H. R. Rao, "Information control and terrorism: Tracking the Mumbai terrorist attack through Twitter," *Inf. Syst. Frontiers*, vol. 13, no. 1, pp. 33–43, Mar. 2011.
[11] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: Real-time event detection by social sensors," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 2010, pp. 851–860.
[12] B. De Longueville, R. S. Smith, and G. Luraschi, "'OMG, from here, i can see the flames!': A use case of mining location based social networks to acquire spatio-temporal data on forest fires," in *Proc. Int. Workshop Location Based Social Netw.*, Nov. 2009, pp. 73–80.
[13] A. Sapienza, S. K. Ernala, A. Bessi, K. Lerman, and E. Ferrara, "DISCOVER: Mining online chatter for emerging cyber threats," in *Proc. Companion Web Conf. Web Conf. (WWW)*, 2018, pp. 983–990.
[14] R. P. Khandpur, T. Ji, S. Jan, G. Wang, C.-T. Lu, and N. Ramakrishnan, "Crowdsourcing cybersecurity: Cyber attack detection using social media," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 1049–1057.
[15] Q. Le Sceller, E. B. Karbab, M. Debbabi, and F. Iqbal, "SONAR: Automatic detection of cyber security events over the Twitter stream," in *Proc. 12th Int. Conf. Availability, Rel. Secur.*, Aug. 2017, pp. 1–11.
[16] K.-C. Lee, C.-H. Hsieh, L.-J. Wei, C.-H. Mao, J.-H. Dai, and Y.-T. Kuang, "Sec-buzzer: Cyber security emerging topic mining with open threat intelligence retrieval and timeline event annotation," *Soft Comput.*, vol. 21, no. 11, pp. 2883–2896, Jun. 2017.
[17] A. Ritter, E. Wright, W. Casey, and T. Mitchell, "Weakly supervised extraction of computer security events from Twitter," in *Proc. 24th Int. Conf. World Wide Web*, May 2015, pp. 896–905.
[18] A. Queiroz, B. Keegan, and F. Mtenzi, "Predicting software vulnerability using security discussion in social media," in *Proc. Eur. Conf. Cyber Warfare Secur.*, 2017, pp. 628–634.
[19] A. Bose, V. Behzadan, C. Aguirre, and W. H. Hsu, "A novel approach for detection and ranking of trendy and emerging cyber threat events in Twitter streams," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2019, pp. 871–878.
[20] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "Mitre ATT&CK: Design and philosophy," MITRE Corp., McLean, VA, USA, Tech. Rep. 19-01075-28, 2018.
[21] B.-J. Koops, J.-H. Hoepman, and R. Leenes, "Open-source intelligence and privacy by design," *Comput. Law Secur. Rev.*, vol. 29, no. 6, pp. 676–688, Dec. 2013.
[22] R. Campiolo, L. A. F. Santos, D. M. Batista, and M. A. Gerosa, "Evaluating the utilization of Twitter messages as a source of security alerts," in *Proc. 28th Annu. ACM Symp. Appl. Comput.*, Mar. 2013, pp. 942–943.
[23] N. Dionísio, F. Alves, P. M. Ferreira, and A. Bessani, "Cyberthreat detection from Twitter using deep neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.
[24] A. Niakanlahiji, J. Wei, and B. Chu, "A natural language processing based trend analysis of advanced persistent threat techniques," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 2995–3000.
[25] G. Ayoade, S. Chandra, L. Khan, K. Hamlen, and B. Thuraisingham, "Automated threat report classification over multi-source data," in *Proc. IEEE 4th Int. Conf. Collaboration Internet Comput. (CIC)*, Oct. 2018, pp. 236–245.
[26] V. Behzadan, C. Aguirre, A. Bose, and W. Hsu, "Corpus and deep learning classifier for collection of cyber threat indicators in Twitter stream," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 5002–5007.
[27] A. Deb, K. Lerman, and E. Ferrara, "Predicting cyber-events by leveraging hacker sentiment," *Information*, vol. 9, no. 11, p. 280, Nov. 2018.
[28] R. Williams, S. Samtani, M. Patton, and H. Chen, "Incremental hacker forum exploit collection and classification for proactive cyber threat intelligence: An exploratory study," in *Proc. IEEE Int. Conf. Intell. Secur. Informat. (ISI)*, Nov. 2018, pp. 94–99.
[29] A. Rakhlin, *Convolutional Neural Networks for Sentence Classification*. San Francisco, CA, USA: GitHub, 2016.
[30] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," 2016, *arXiv:1603.01360*.
[31] E. Akleman, "Deep learning," *Computer*, vol. 53, no. 9, p. 17, Sep. 2020.
[32] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 160–167.
[33] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, Jul. 1997.
[34] X. Liu, P. He, W. Chen, and J. Gao, "Multi-task deep neural networks for natural language understanding," 2019, *arXiv:1901.11504*.
[35] J. Baxter, "A Bayesian/information theoretic model of learning to learn via multiple task sampling," *Mach. Learn.*, vol. 28, no. 1, pp. 7–39, Jul. 1997.
[36] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[37] J. Robertson, A. Diab, E. Marin, E. Nunes, V. Paliath, J. Shakarian, and P. Shakarian, *Darkweb Cyber Threat Intelligence Mining*. Cambridge, U.K.: Cambridge Univ. Press, 2017.
[38] *Standard Twitter Streaming Api Request Parameters*, Twitter Developer, San Francisco, CA, USA, 2021.
[39] *Command and Control Server*, Trend Micro, Tokyo, Japan, 2021.
[40] A. Mansouri, L. S. Affendey, and A. Mamat, "Named entity recognition approaches," *Int. J. Comput. Sci. Netw. Secur.*, vol. 8, no. 2, pp. 339–344, 2008.
[41] *Dbpedia About*, DBpedia, Leipzig, Sachsen, Germany 2021.
[42] M. F. Porter, *An Algorithm for Suffix Stripping*. MCB UP Ltd, 1980.
[43] B. Bengfort, R. Bilbro, and T. Ojeda, *Applied Text Analysis With Python: Enabling Language-Aware Data Products With Machine Learning*. Sebastopol, CA, USA: O'Reilly Media, 2018.
[44] J. Ramos, "Using TF-IDF to determine word relevance in document queries," in *Proc. 1st Instructional Conf. Mach. Learn.*, vol. 242, no. 1, 2003, pp. 29–48.
[45] K. Pasupa and W. Sunhem, "A comparison between shallow and deep architecture classifiers on small dataset," in *Proc. 8th Int. Conf. Inf. Technol. Electr. Eng. (ICITEE)*, Oct. 2016, pp. 1–6.
[46] C. F. Pampel, *Logistic Regression: A Primer*. Newbury Park, CA, USA: Sage, 2020.
[47] D. C. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008.
[48] W. Zhou, L. Zhang, and L. Jiao, "Linear programming support vector machines," *Pattern Recognit.*, vol. 35, no. 12, pp. 2927–2936, 2002.
[49] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[50] T. Pranckevičius and V. Marcinkevičius, "Comparison of naive Bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic J. Modern Comput.*, vol. 5, no. 2, 2017.
[51] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1997.

RENATO MARINHO was born in Fortaleza, Brazil, in 1979. He received the B.S. and M.S. degrees in applied informatics from the University of Fortaleza, Brazil, in 2004 and 2013, respectively.

Since 2013, he has been teaching post-graduate malware analysis and incident response disciplines. Since 2015, he has been the Chief Research Officer with Morphus Labs, where he conducts studies about cyber threats capabilities and defensive methods. Since 2017, he has been an Incident Handler with the SANS Internet Storm Center, where he conducts threats analysis to collaborate with the community as a volunteer position. His research interests include cyber threat visibility measurement, emerging cyber threat discovery, and malware analysis.

RAIMIR HOLANDA was born in Fortaleza, Brazil, in 1967. He received the five-year B.Sc. and M.Sc. degrees in computer science from the Federal University of Ceará and the Ph.D. degree in computer engineering from the Technical University of Catalonia (UPC).

He held a postdoctoral position with Pierre and Marie Curie University. He is currently the Leader of the Sensor Networks Laboratory Research Group, University of Fortaleza. He has more than 100 publications in international journals, conferences, and book chapters. His main research interests include the IoT and sensor networks, blockchain, and cybersecurity.