BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **22**, No 2
Sofia • 2022 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2022-0019

Visualizing Interesting Patterns in Cyber Threat Intelligence Using Machine Learning Techniques

Sarwat Ejaz[1], *Umara Noor*[1], *Zahid Rashid*[2]

[1]*Department of Computer Science and Software Engineering, International Islamic University, Islamabad, Pakistan*
[2]*Technology Management Economics and Policy Program, College of Engineering, Seoul National University, 1 Gwanak-Ro, Gwanak-Gu, 08826, Seoul, South Korea*
E-mails: sarwatijaz68@yahoo.com umara.zahid@iiu.edu.pk rashidzahid@snu.ac.kr

Abstract: *In an advanced and dynamic cyber threat environment, organizations need to yield more proactive methods to handle their cyber defenses. Cyber threat data known as Cyber Threat Intelligence* (*CTI*) *of previous incidents plays an important role by helping security analysts understand recent cyber threats and their mitigations. The mass of CTI is exponentially increasing, most of the content is textual which makes it difficult to analyze. The current CTI visualization tools do not provide effective visualizations. To address this issue, an exploratory data analysis of CTI reports is performed to dig-out and visualize interesting patterns of cyber threats which help security analysts to proactively mitigate vulnerabilities and timely predict cyber threats in their networks.*

Keywords: *Cyber threat intelligence, machine learning, visual analytics, tactics techniques and procedures, cyber threat actor, malware.*

1. Introduction

In the fourth industrial revolution era, the individuals and states are heavily relying on the implementations of information technologies in almost every area as compared to the past. The electronic devices including smartphones, TVs, and home appliances (such as air conditions, refrigerators, light bulbs, and cooking ranges, etc.) are getting smarter at a very fast pace. The increased reliance on IT is giving rise to a range of cyber security threats posed on both individuals and at state level. Recent survey shows [1], that the top target of attackers is industry encompassing high profile business tycoons and financial sectors. Similarly, the governmental and the individual targets are on second and third priority respectively depending on the motivations of the attackers [2]. Combating such cyber-attacks need a timely response in the first place as attackers are innovating at much faster pace than the defenders. Malwares are commercializing in the form of attack kits and are readily available on the underground forums. Botnets are also available on rent to launch automated cyber-attacks. Despite all these facts, the patterns of attacks can be

identified as the attackers are reusing attack techniques, malwares, command and control protocols during attack campaigns. Therefore, the probability is very high that an organization or group who has already faced an attack can share cyber threat information about that attack along with its mitigations to its trusted partners. This capability is termed as CTI sharing. CTI is now becoming a vital part of an organization's cyber security defense set-up. The term "intelligence" refers to analyzed and actionable information. The CTI can be obtained from organization's internal sensors as well as from external sources in the form of cyber threat feeds. CTI plays an important role in analyzing threat sources and applying appropriate defenses against them. The obtained CTI can be shared with the trusted partners in order to provide resilience and proactive defense against sophisticated cyber-attacks.

In past few years, an inclined trend has been observed for putting efforts related to managing CTI and sharing it within trusted communities. The main objective behind this trend is to make the cyber defense mechanisms proactive instead of reactive. To enable organizations to have this level of CTI capability, many standards and tools have been developed. Some notable ones are Traffic Light Protocol (TLP) [3], Open Indicators Of Compromise (OpenIOC) framework [4], Vocabulary for Event Recording and Incident Sharing (VERIS) [5], Incident Object Description and Exchange Format (IODEF) [6], Cyber Observable eXpression (CybOX) [7], Structured Threat Information eXpression (STIX) [8] and Trusted Automated eXchange of Indicator Information (TAXII) [9].

The volume of CTI is increasing very rapidly. There are several open source CTI repositories available [10], such as Hail-a-Taxii [11], and Adversarial Tactics Techniques & Common Knowledge (ATT&CK) [12]. They are famous for sharing indicators of compromise and cyber threat actor's attack patterns respectively. At the time of writing, there are more than 0.11 million indicators of compromise reported in Hail-a-Taxii [11] which makes it a significant issue for big data analytics [13]. Similarly, there are 185 Tactics, Techniques and Procedures (TTP) of 122 cyber threat attackers available in ATT&CK [12].

The tools available in the market for visualizing CTI data have very limited capabilities. It is not possible for security analyst to visualize massive CTI data from multiple perspectives in a flexible manner and can support in applying strong cyber defenses. One of the challenge for security analyst is to analyze huge volume of CTI and utilize it effectively to make it actionable intelligence. Based on this challenge, the research problem addressed in this research work is to provide visual analytics of CTI from multiple perspectives using machine learning techniques and to find interesting patterns that can be leveraged to launch proactive and actionable defense against emerging cyber-attacks.

Based on the aforementioned research problem, the first objective of this research work is to facilitate the cyber security analyst in understanding the current massive cyber threat landscape and applying appropriate mitigations. The second objective is to check which machine learning technique effectively predicts cyber threat actor and malware in CTI data

To find interesting patterns, an exploratory data analysis is performed on open source dataset. The results are shown in the form of visualizations that can help in

their networks. To predict cyber threat actor or malware, machine learning models are trained using attack patterns, also known as TTP. The trained models help in identifying the culprit or malware behind the cyber incident. The experimental results show that our proposed approach help security analyst effectively analyze CTI data and predict cyber threat actors and malware with 86% accuracy.

The rest of the research paper is organized as follows: Section 2 discusses about the standards used to store and visualize CTI, and open source repositories of CTI. Section 3 presents the related work in the domain of CTI, and review of machine learning techniques to dig out interesting patterns from CTI. Section 4 presents the proposed solution. Section 5 describes the experiment and evaluation of our proposed solution. The research work is finally concluded in the Section 6.

2. Standards and open source repositories of CTI

This section presents a detailed background related to CTI standards such as STIX [14], Hail-a-Taxii [11], and ATT&CK [12] and other open source repositories of CTI.

MITRE is a research and development center and it is working on development of various standards, such as, STIX, ATT&CK, TAXII, Cyber Observable expression (CybOX), Malware Attribute Enumeration and Characterization (MAEC). These standards provide sharing and management of CTI by resolving the issues of interoperability among organizations.

STIX is a standardized language developed by MITRE [14] and is adopted as an international standard by various intelligence sharing communities and organizations. It has been designed to be shared via a secure protocol named Trusted Automated eXchange of Intelligence Information (TAXII) [9]. However, other ways are also available for sharing STIX data. STIX describes cyber threat information as observable, indicator, incident, TTP, exploit target, campaign, threat actor and Course Of Action (COA). Observable is used to represent a static or dynamic event. An indicator is extension of the observable. It describes the context of an event. The context can be time, information source, IP address, file hashes and domain names. An incident describes the context of activities associated with an incident and adversary. The TTP are the attack patterns of the attacker using which he launches an attack. The exploit target is the victim resource of the organization. It is often related to the vulnerability in the resource. If the CTI is about a particular cyber threat actor, then the campaign construct may contain a list of cyber incidents associated with that particular adversary. Threat actor describes attributes of the adversary. The COA are the steps related to the mitigation of the cyber threat.

TAXII [9] defines how cyber threat information can be shared via services and message exchanges. TAXII is becoming an international standard. It is designed specifically to support STIX information, which it does by defining an API that aligns with common sharing models. There are three principal models for TAXII: 1) hub and spoke, 2) source/subscriber, 3) peer to peer. In hub and spoke model, there is one central repository of information. The hub is the central information resource. The spoke can be both consumer and the information provider. In source/subscriber model

there is one single source of information. In peer to peer model, there is no central authority, multiple groups can share information.

Hail-a-Taxii [11] is an online repository of CTI having data in STIX format. At the time of writing, it has more than 1107066 indicators in the repository currently.

Another CTI repository known as ATT&CK [12] 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. ATT&CK data help the security analyst in understanding the threat actor and the possible courses of action to mitigate its attack patterns known as TTP. Most common job of a security analyst is to identify the common behavior amongst the threat actors to devise a common course of action. If the TTP of a cyber-threat actor are determined a powerful defense can be devised against it. A structured format is provided by ATT&CK which allows to categorize threat behaviors. It also provides attack pattern, tactics, malware and tools which are used for classification and visualization of the threat information.

CTI provide information about recent cyber threats and cyber threat actors in separate segments. All the segments are thoroughly discussed by S t r o m et al. [15]. It includes various fields like Tactics, Techniques and Procedures (TTPs), software's, behaviors, methods and procedures and malware information.

Other open source repositories for threat intelligence are IBM X Force [16], Symantec [17], FireEye [18], and CrowdStrike [19]. These repositories describe cyber threat data with different levels of abstraction using different formats.

In view of above, may be concluded that cyber threat data is available for analysis to the security community in different open source repositories but, due to its huge mass there is a need to dig-out interesting patterns that can be leveraged to make cyber defenses strong.

3. Machine learning and visualization of CTI

This section encloses a detailed study of prior work done in the domain of CTI related to the visualization of cyber threat information, and a review of machine learning techniques to dig out interesting patterns from it.

The importance of CTI is highlighted in [15] which states that the use of threat data by organizations is rapidly increasing. CTI provides valuable insight about the malware, attacker and its mitigation. Various organizations are investing in CTI to proactively defend cyber-attacks. However major issues regarding the mass of CTI, search for interesting patterns and their automated consumption in security controls, such as, firewall, Intrusion detection system, and honeypots requires attention from the research community. CTI encompasses information about both internal and external threats. An effective insight into external threats can help protect organization from harm with more gravity.

Currently the tools available for visualizing CTI data have limited capabilities. It is not possible for security analyst to visualize massive CTI data from multiple

perspectives and flexibility that can help in applying strong defenses. Bromiley [20] propose STUCCO which is a cyber-security knowledge graph which has lots of levels of abstractions in cyber security. The purpose is to collect information from 13 sources which include structured as well as unstructured data which is established in a common format which could be used to assist security analyst and malware analysis tools. The ontology has been implemented in JSON and it is also compatible with Graph JSON format. One major limitation of this tools, is that it does not provide CTI visualization tools. In addition, the malware entity from the ontology only tells about the form of attack and does not provide details how it was launched. Further, the behavior and the tactics are also not reported in the in the proposed ontology.

A web-service has been proposed by Craig et al. [21] for malware analysis. This web-service provides the capability of malware analysis based on its code and semantics. It takes the help of the interactive graph to find the relationship among malwares. This relationship visualization helps the security analyst in malware classification. The proposed web service does not provide in depth CTI data analysis.

Another tool for the CTI visualization is commercially available called STIXviz [22]. It is a Javascript-based tool used for the visualization of STIX documents. STIXViz offers three kinds of visualizations for STIX documents: 1) timeline view, 2) graph view, and 3) tree view. The timeline view displays the entities according to the recorded date and time in a STIX document, such as incidents and their related campaigns. The graph view dynamically places the nodes by using force directed graph design. Nodes can be moved to a new location; in this case the design of the graph will dynamically rearrange itself. The tree view displays STIX entities at their top or first level: entities include observables, indicators, TTPs, threat actors, campaigns, course of actions, exploit targets and incidents. When the tree expands, it first displays a top-level node which is shown for each element category that the STIX file contains. On clicking the node with black and white border it is expanded and displays second level information. A major limitation of the STIX VIZ design tool is that it collapses when the CTI data become large, as shown in Fig. 1.

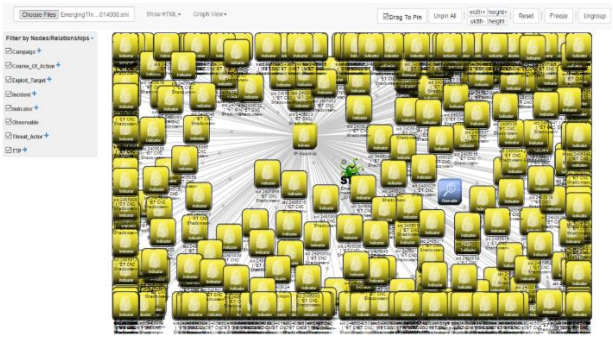

Fig. 1. STIX VIZ visualization collapses for large amount of data

Steven Noel from the MITRE Corporation in his paper defines various interactive visualization approaches for knowledge base of Common Attack Pattern Enumeration and Classification (CAPEC) [23]. The proposed visualizations analyze CAPEC attack pattern taxonomy, and displays the hierarchical relationships of attack

classes and subclasses. There are three interactive visualizations to display CAPEC taxonomy. The first one is the sunburst visualization [24], which draws diverge lines from a center for all the levels of the tree. The second one is the circular tree map [25]. It provides huge display areas for higher levels of the tree, thus emphasizing on the more general attack patterns. The third one is the Voronoi tree map [26], which repeatedly divide the screen space and possesses balanced interactive display as per unity aspect ratio. These visualizations are very interactive and useful in utilizing the screen space efficiently to display large amounts of hierarchical data. Based on the usefulness of these tree map visualizations, we employ them to display ATT&CK knowledge base and malware information.

In another work, Z h a o and L u [27] proposed a different circular tree map for the efficient utilization of screen space. It efficiently maximize screen space utilization ratio as compared to traditional circular tree map. It inherits the best features of circular tree map for hierarchical structure as well as improves and optimizes the screen space utilization. They also provide a comparison of the three widely accepted tree maps: 1) Voronoi, 2) Squarified and 3) Circular. They found that the circular tree map better reveal the hierarchical structure than other two tree maps. Then they also compared circular tree map with their improved circular tree map. It is shown that the optimized circular tree map utilizes screen space more efficiently.

D a n i e l, E n d e r t and K i d w e l l [28] discuss the major challenges that should be considered during the designing and implementing of network security visualizations. They describe how challenges are met by providing an example of a prototype named as “SEQViz”. It facilitates two views for coordination, the model view and the network overview view. Both views enable higher level of detail to be displayed for the view of interest by expanding and collapsing while retaining the other view in frame of reference. It is an effort to increase the adoption and acceptance rate of visualization approaches and tools in network security. The views for coordination by SEQViz allow the users to view and manage the large amount of available information. SEQviz is a prototype visualization tool for network security. It is not possible to visualize CTI using SEQviz.

An online survey has been conducted by Nick et al. to measure the aesthetics, efficiency and the effectiveness of all different types of data visualization techniques used for the representation of hierarchical datasets [29]. A comparison of all visualization techniques is done according to the completion speed, accuracy, latency of incorrect feedback and function termination. After critical evaluation of all techniques, sunburst visualization is ranked on the top among others based on aesthetics, efficiency and effectiveness.

B r o n w y n, P e r l and L i n d a u e r [30] discuss that using machine learning techniques unstructured data can be structured. They revealed it by a case study of actual incident data. They identified a major question: given a large collection of shared incident information, “how a community can get value from it?” As a result, they proposed a solution to combine such low-quality unstructured incident information with machine learning methods to make such information more consumable and increase the participation in the information sharing community.

They stated that, by using machine learning techniques, one can find more information related to indicators and incidents of shared data in order to complete the incident information for analysis and decision making. But they applied highly unsupervised (clustering) machine learning techniques. For malware detection and malware family identification, classification techniques give more accuracy as compared to clustering techniques in all types of analyses (static, dynamic and hybrid) [31].

In a research work, Z a h r a et al. [32] review existing heuristic based malware detection approaches such as signature-based and behavior-based. They identified that signature-based detection approaches are unable to detect new malware and behavior-based approach is used to overcome this problem, but the limitation of this approach is non-availability of promising false positive ratio and requires the high amount of time for detection. To overcome the disadvantages of these approaches, heuristic-based detection technique has been introduced. It uses machine learning techniques to understand the behavior of the executable file. The features used for heuristic-based detection are API calls, CFG (Control Flow Graph), N-Gram, Opcode (operational code) and hybrid features. They also have given the brief overview of advantages, disadvantages, and features. According to their review, the major disadvantage of heuristic detection is the high false positive ratio.

K y l e et al. [33] define, that cyber warfare is a back-and-forth fight between threat groups and defenders. They explained the struggle by demonstrating two case studies of freely available information for the Black Energy and Zeus malwares over the last eight years. This kind of relationship is recognizable as malwares such as Black Energy and Zeus continuously succeed to conquer defensive abilities.

The changes in malware throughout the past several years is explored in malware trends [34]. The emphasis is to identify what is most likely seen by security industry nowadays; how the organizations can strengthen their systems and networks to prevent attacks, and the awaited targets and developments in the next coming years. The paper describes malware changes and attacker tactics changes in order to prevent systems. It briefly describes the currently popular malware like ransomware which is used to attack cryptocurrencies such as bitcoin, which allow untraceable fund transfer. The goal of ransomware is to gain access of user system, their personal information and encrypting everything and then makes a demand, usually to transfer money through cryptocurrency in the specified time, otherwise their data will be permanently unrecoverable.

In another work, a malware dataset of Zeus banking Trojan was developed by A b e d e l a z i z and A l r a w i [35]. The system captures the multiple artifacts or features (file system, registry, IP, network protocol, network connections, etc.) about a given malware samples. Then they classify these features automatically by using four different machine learning approaches: K-Nearest Neighbor (KNN), Support Vector Machine (SVM), decision trees and logistic regression for classification of Zeus Malware. They identified 65 unique and vigorous features for recognizing malware. It is binary classification dealing with two classes of files, either the malware is malicious or not. To detect the family of the malware, they use features of file system, registry, and network attributes. The machine learning models are

evaluated by measuring both the efficiency and accuracy. According to their evaluation results, KNN provides more accurate results as compared to others. Also, it is simple to implement, and it gives still better and reasonable results even in biased situation or less demonstrative training dataset. The limitation is that it does not identify the malware family. XGBoost ensemble learning model of deep neural network have been used for anomaly detection in intrusion detection systems [37].

The aforementioned literature review reveals a significant research gap in the domain of exploratory data analysis for CTI. Also, there is a need to evaluate the prediction capability of machine learning techniques for the currently available CTI data.

4. Proposed solution

The research work discussed here comprise of two tasks: 1) find interesting patterns, an exploratory data analysis is performed on Hail-a-Taxii. 2) predict cyber threat actor and malware based on their TTP using machine learning models.

In Fig. 2, the steps of our proposed solution are shown. In the first step both datasets are downloaded. In the second step the data is pre-processed to be used for exploratory data analysis and malware prediction. Hail-a-taxii dataset can be downloaded using Anomali STAXX [38] and python script. Anomali STAXX provide limited information along with a user interface to view data directly. The downloaded data is in JSON/CSV format so there is no need of parser. Python script, on the other hand provides complete information in XML format without a user interface. There is a need of a parser to convert data in CSV format. To predict cyber threat actor and malware, TTP dataset of N o o r et al. [39] has been used. This dataset is collected data from five sources, i.e., ATT&CK [12], IBM X-Force [16], Symantec [17], FireEye [18] and CrowdStrike [19]. The details of datasets download and their pre-processing challenges are discussed in Section 5.1.

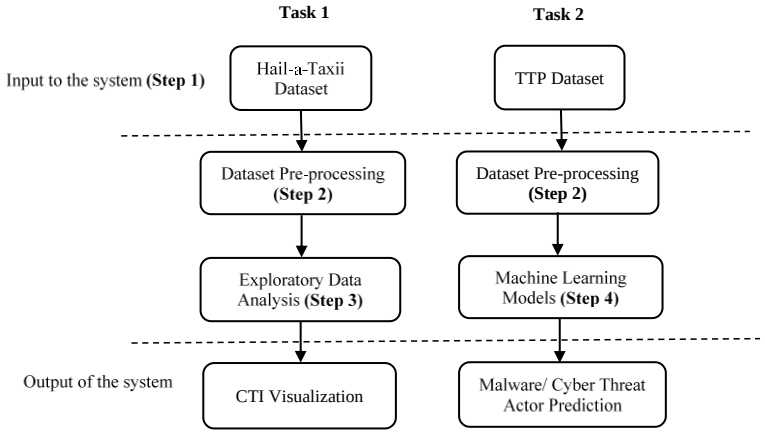

Fig. 2. Architecture diagram of the proposed system

In the third step, exploratory data analysis of Hail-a-Taxii dataset is performed which results in its effective visualizations from multiple perspectives. The purpose

of exploratory data analysis of CTI is to enable security analysts understand their current massive cyber threat landscape and apply targeted mitigations in an effective way. The details of exploratory data analysis are provided in Section 5.2.

In the fourth step machine learning models are built to predict cyber threat actors and malware family from the TTP dataset. The purpose of building machine learning models is to predict the cyber threat actor and malware based on their behavioral indicators of compromise, i.e., TTP. TTP reflect the tactical attack strategies of a cyber threat actor which is an integral part of their training and are hard to change. The details of building machine learning models and threat actor and malware prediction are given in Section 5.3.

The machine learning model is trained using different techniques, i.e., KNN, Naïve Bayes, Naïve Bayes (Kernel), decision tree, random forest, gradient boosted trees, Artificial Neural Network (ANN), deep learning, generalized linear model, linear regression (using ensemble classification) [34], and ensemble learning. The results of these machine learning techniques are discussed in Section 5.3.

5. Experiment and evaluation

In this section, the details of dataset download, pre-processing, exploratory data analysis results and machine learning prediction results are discussed.

5.1. Datasets download and challenges

Hail-a-Taxii dataset can be downloaded using anomali STAXX [38] and python script. There are a few differences between these two approaches. Anomali STAXX provide user interface to view data directly. It provides limited information. The downloaded data is in JSON/CSV format so there is no need of parser. Python script doesn't provide user interface to view data directly. It provides complete information in XML format. The data needs a parser for conversion in CSV format.

To get CTI using Anomali STAXX, its client module is downloaded. The service is specified as hail-a-taxii and it is switched on all polls to start collecting data. The dataset is downloaded in CSV format by running all feeds poll.

To get CTI using python script, http request is sent to hail-a-taxii server and the feeds are downloaded from the server. This method gives more columns and detailed data than the Anomali STAXX client. However the data needs to be converted into CSV format. Also a major challenge faced with CTI data download using python script was the memory error. An 8GB RAM was used whereas the data to be downloaded was a lot more than this. For this purpose, Google Cloud Platform servers were used to create powerful instances to download most of the feeds.

To obtain a common CSV header of hail-a-taxii data using the python script, we had to parse the file by getting useful information from each content block of indicator, observable and TTP, these all combine to provide information of a single TTP or malware. In this way a strategy was devised to keep the atomicity of the dataset.

In order to predict cyber threat actor and malware, we have used dataset of Noor et al. [39]. This dataset is collected data from five sources, i.e., ATT&CK [12], IBM X-Force [16], Symantec [17], FireEye [18] and CrowdStrike [19]. The ATT&CK dataset can also be separately downloaded from open-source GitHub link. This data is available in JSON format, it is parsed in CSV format for further analysis.

ATT&CK data is in JSON format which needs to be parsed into a CSV file. Data was arranged in a way that each object in the array was corresponding to one of the malwares, techniques, course of action, tools or relation between one of them. We successfully parsed this data using PHP script to get a CSV file with malwares as class labels and techniques as attributes. After parsing, data set looks like Table 1, where 1 and 0 represent the presence and absence of a TTP for a particular cyber threat actor or malware. The class to be predicted is labelled as "cyber threat actor or Malware". The security community attribute cyber threats towards their threat actors or the malware used in the incident. The rest of the columns represent TTP related to cyber threats. Mimikatz and Pwdump are the examples of software tools used by the attacker.

Table 1. Parsed Dataset of ATT&CK MITRE [12], IBM X-Force [16], Symantec [17], FireEye [18], and CrowdStrike [19]

No	Cyber threat actor/Malware	TTP 1	TTP 2	TTP3	…….	Mimikatz	Pwdump	…….
1	Deep Panda	0	0	0	…….	1	0	…….
2	Drago0k	1	0	0	…….	1	1	…….
3	Dragonfly	1	1	0	…….	0	0	…….
4	Dust Storm	0	1	0	…….	1	1	…….
5	Equation	0	1	1	…….	1	0	…….
6	Fin6	0	0	0	…….	1	0	…….
7	GCMAN	1	0	0	…….	1	0	…….

5.2. Exploratory data analysis

The dataset downloaded using Anomali STAXX contains 16 attributes: 1) indicator, 2) classification, 3) confidence, 4) itype, 5) type, 6) severity, 7) tlp, 8) source, 9) feed_site_netloc, 10) feed_name, 11) detail, 12) date_last, 13) actor, 14) campaign, 15) ID, 16) recid.

After reviewing the data, it is found that the actor and campaign attributes were empty and useless. And there are three attributes, i.e., classification, tlp, and feed_site_netloc that were 100% stable meaning that they had only one value in the whole column, so they were also discarded. Lastly, the indicator, ID & recid fields were high in cardinality so they were also discarded for the next step of classification. All these steps are performed to clean and transform the data for further analysis.

The dataset downloaded using python script contains 36 attributes: 1) package ID, 2) package timestamp, 3) indicator ID, 4) indicator timestamp, 5) indicator title, 6) observable type, 7) indicator types, 8) TTP type, 9) TTP name, 10) domain, 11) domain is FQDN, 12) URI, 13) URI type, 14) IP, 15) IP category, 16) IP is source, 17) port, 18) protocol, 19) file, 20) file hash, 21) file format, 22) file hash type, 23) producer name, 24) indicator description, 25) observable description, 26) TTP description, 27) observable ID, 28) indicator observable ID, 29) TTP ID, 30) TTP

timestamp, 31) TTP malware instance ID, 32) indicator TTP ID, 33) produced time, 34) received time, 35) contributing sources, 36) references.

Among the above attributes observable ID, indicator observable ID, TTP ID, TTP timestamp, TTP malware instance ID, and indicator TTP ID were discarded due to high cardinality.

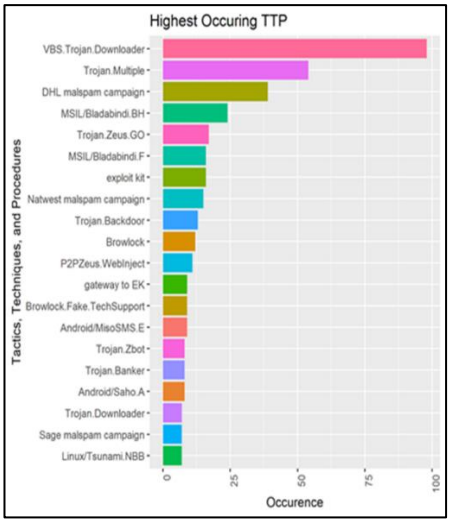

Fig. 3. Frequency of occurrence of TTP

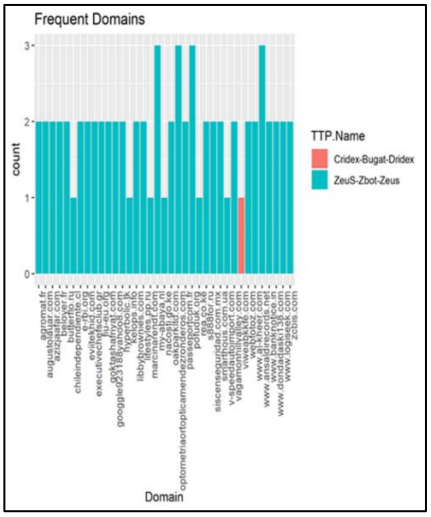

Fig. 4. Frequency of occurrence of domain names

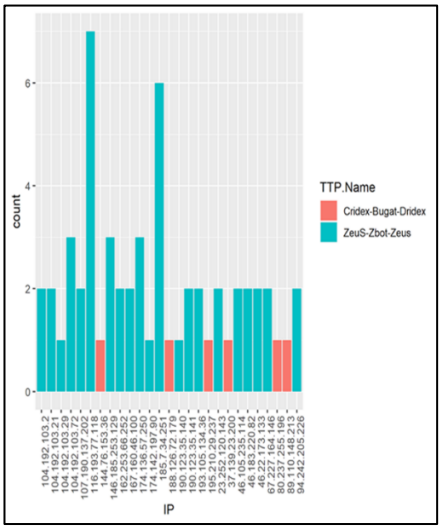

Fig. 5. Frequency of occurrence of IPs

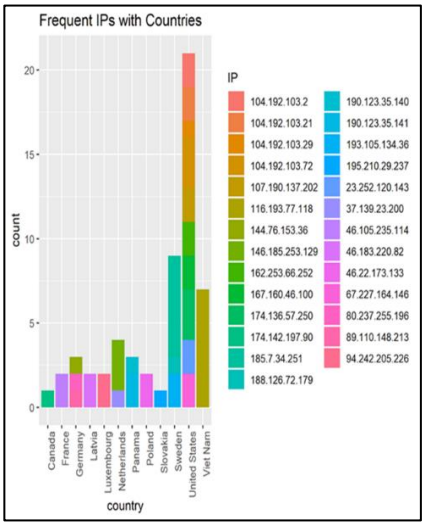

Fig. 6. Country-wise IP frequency

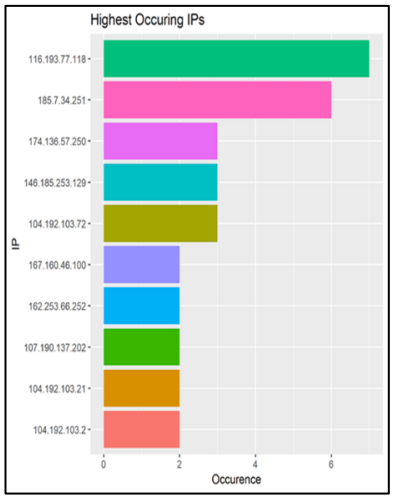

Fig. 7. IP Frequency for a segment of dataset

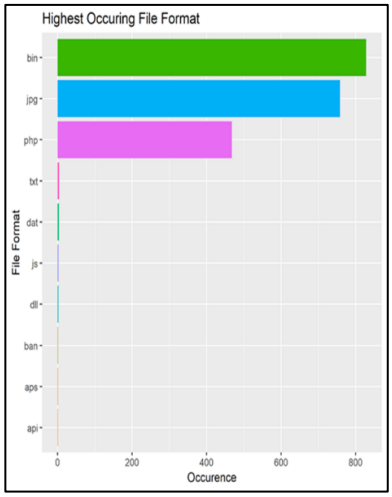

Fig. 8. Frequency of malicious file formats

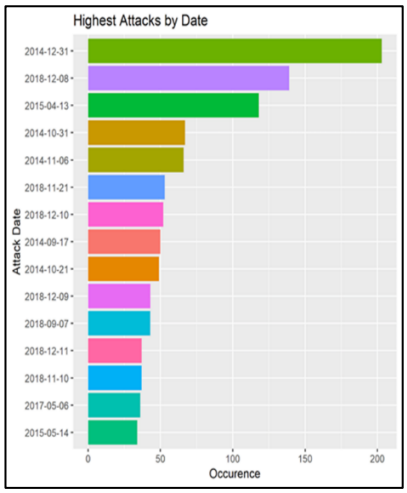

Fig. 9. Temporal analysis of cyber attacks

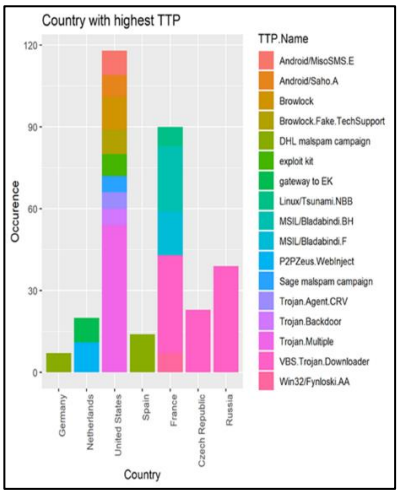

Fig. 10. Country wise frequency of TTP

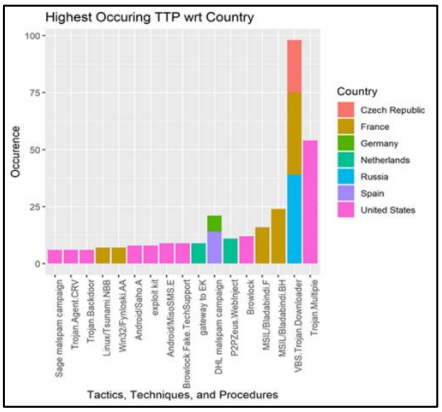

Fig. 11. Highly frequent TTP w.r.t. country

In the following, the visualizations of our experiment are discussed. The coding was done using R programming language. The complete code of the experiment is provided to the research community for further analysis [40].

In Fig. 3, the result of exploratory data analysis of hail-a-taxii dataset is shown. It shows the frequency of occurrence of each TTP. It can be seen that VBS.Trojan.Downloader has the highest frequency of occurrence. This data visualization helps the security analyst to know about the most frequently occurring TTP over the past years in the massive CTI data. In Fig. 5, the hail-a-taxii dataset is analyzed from another perspective. It shows the frequent domains used with the two TTPs Cridex and Zeus. It can be seen that the dataset has more domains related to Cridex TTP as compared to Zeus. Dataset visualization from this perspective can help security analyst to locate more frequently used domains used by attackers to launch attacks. This data can be used to block the frequent as well as single occurrence domains in the host or network based security firewalls. Another view of the hail-a-taxii dataset is provided in Fig. 6. It shows the frequent IP addresses used with the two TTPs Cridex and Zeus. It can be seen that the dataset has more domains related to Zeus TTP as compared to Cridex. Dataset visualization from this perspective can help security analyst to locate more frequently used IPs used by attackers to launch attacks. This data can be used to block the frequent as well as single occurrence IP addresses in the host or network based security firewalls. The IP frequency with respect to country is shown in Fig. 7. On the right hand side there are IP addresses represented by different colors. On the left hand side, the country-wise usage of these particular IPs is shown. It can be seen that there are certain IPs which are used in different countries to launch attacks, such as, 193.105.134.36 is used in Panama, Slovakia, and Sweden. It shows that there is a possibility of a common cyber threat actor. According to Fig. 7, the highly frequent IP is 116.193.77.118 and it is used for attacks in Vietnam. Based on this data, the security analyst can block IPs from his own country as well as from other countries. The security analyst can also analyze frequent IPs in a particular segment of CTI as shown in Fig. 8. To launch attacks via malware different file formats are used. Their usage trend also changes from time to time. In Fig. 9, the frequency of file formats used for malicious transfers is shown. It can be seen that binary files are more frequently used for transferring malware in the Hail-a-Taxii dataset. The time-wise cyber-attack occurrence is shown in Fig. 10. It can be seen that December, 2014 was the peak time of cyber-attacks. The TTP frequency with respect to country is shown in Fig. 11. On the right hand side there are TTP represented by different colors. On the left hand side, the country-wise usage of these particular TTP is shown. It can be seen that there are certain TTP which are used in different countries to launch attacks, such as, VBS.Trojan. Downloader is used in France, Czech Republic, and Russia. It shows that there is a possibility of a common cyber threat actor in these attacks. According to Fig. 11, the highly frequent TTP is Trojan.multiple and it is used for attacks in USA. Based on this data, the security analyst can apply mitigations for TTP common in his country as well as in other countries. Another view of country-wise TTP frequency is shown in Fig. 12.

In view of above, it is concluded that the proposed exploratory of CTI provides an effective way of understanding the current massive cyber threat landscape and

applying targeted mitigations as compared to the existing CTI visualization approaches [22, 23].

5.3. Cyber threat actor/ malware prediction

After exploratory data analysis of Hail-a-Taxii dataset, the machine learning models are built using the TTP dataset [39]. It is a very rich dataset encompassing TTP of cyber threat actors and malware in phases of cyber kill chain model. Cyber kill chain describes steps used by the attacker to launch a malicious activity. The attributes in the dataset are TTP, software tools and cyber threat actors/malware. Cyber threat actor/malware is the class label which will be predicted based on the TTP and software tools. Cyber threat actor also encompasses malware kind.

As shown in Table 1, the class label of the dataset is cyber threat actor/malware. The TTP are the features or attributes which shows its existence with respect to class label. The dataset is normalized to remove zero entries by adding 1 to all features. For training machine learning model, the dataset was split into 70% and 30% ratio to get training and testing data, respectively. Cross validation was applied to divide dataset into multiple samples. The format of feeding the data to the model and then extracting data remains the same for each model as mentioned. The results of accuracy, precision, recall, F-Measure, and execution time are shown in Table 2. The results show that among the single base machine learning models, random forest is predicting cyber threat actors/malware with a high accuracy (82%), precision (81%), recall (75%), and F-Measure (0.78).

To improve the prediction results, ensemble learning models, i.e., voting, bagging, boosting, and stacking are trained. The results in Table 2 show that ensemble learning models outperformed the base machine learning models.

The purpose of ensemble learning models is to combine weak machine learning base models to develop an optimal and effective predictive solution. An effective solution is achieved by reducing variance of certain base learners, such as, neural networks. The ensemble learning techniques that are used are: voting, bagging, boosting, and stacking.

In voting, multiple models make prediction about a class. These predictions are called “vote”. The final prediction is determined by vote from majority of the models. The voting model was built using random forest, KNN, and Naïve Bayes. These three base models were selected based on their high predictive performance among the rest of the base models. The result shows that voting model is predicting cyber threat actors/malware with a high accuracy (84%), precision (81%), recall (79%), and F-Measure (0.78).

In bagging, the results of multiple models of a single base learner are combined, such as, decision tree to get a final prediction. Bagging is also known as parallel ensemble. The base learners work in parallel during the training process. Here the bagging model is built with random forest. The reason for using random forest is its high prediction performance as a single base learner. The result shows that bagging model is predicting cyber threat actors/malware with an accuracy (82%), precision (78%), recall (75%), and F-Measure (0.77). The predictive performance of this model is low as compared to voting model.

As compared to bagging, boosting is a sequential ensemble. The base learners are sequentially trained to get a final prediction. In this paper, AdaBoost, and gradient boosting tree are used. The predictive performance of gradient boosting tree is not very promising. However, AdaBoost have an accuracy (83%), precision (76%), recall (79%), and F-Measure (0.78). The predictive performance of the Adaboost model is low as compared to the voting model however it is high as compared to bagging and gradient boosting trees.

In stacking, different learners are used. The learners can be base learners (decision tree, KNN, or Naïve Bayes) or they can use other ensemble learning models. In this paper, four different stacked models are build. The first model is build using two instances of random forest. The second model is built using a Naïve Bayes (kernel) and a random forest. The third model is built using a voting model and a random forest. The voting model is built using KNN, Naïve Bayes, and random forest. The fourth model is built using a voting and a bagging model. The voting model of the fourth model is the same as the previous one, while bagging model is built using random forest. Among all these stacking models and the rest of the machine learning models, the fourth stacking model has the highest predictive performance. The accuracy is 86%, precision is 79%, recall is 79%, and F-Measure is 0.79.

The execution time of all the models is also observed. The results show that the voting model with an execution time of 10 s is more efficient as compared stacking model with an execution time of 1 minute and 15 s.

Based on the results, it is concluded that random forest as a single base learner is best suited for the TTP dataset. When considering ensemble learning models stacking that combines voting and bagging is more effective with respect to prediction as compared to other models.

Table 2. Prediction results of machine learning techniques

No	Machine learning technique	Accuracy (%)	Precision (%)	Recall (%)	F-Measure	Execution time (min: s)
1	KNN	71.97	74.21	66.48	0.7	0:01
2	Naïve Bayes	78.79	76.41	72.92	0.75	0:01
3	Naïve Bayes (Kernel)	79.55	76.67	73.67	0.75	0:01
4	Decision Tree	9.85	7.31	6.89	0.07	0:01
5	Random Forest	81.82	81.49	75.38	0.78	0:05
6	Gradient Boosted Trees	18.94	16.28	15.53	0.16	2:10
7	Artificial Neural Network	34.09	21.89	23.9	0.23	0:01
8	Deep Learning	63.64	57.61	56.44	0.57	0:05
9	Generalized Linear Model	75	72.76	68.18	0.7	0:08
10	Linear Regression	59.09	55.72	56.25	0.56	0:01
11	Ensemble Voting (Random Forest, KNN, Naïve Bayes (Kernel))	84.07	**80.73**	78.60	**0.79**	**0:10**
12	Ensemble Bagging (Random Forest)	81.87	78.02	75.38	0.77	1:00
13	Ensemble AdaBoost (Random Forest)	82.64	75.95	**79.27**	0.78	0:23
14	Ensemble Stacking (Random Forest, Random Forest)	82.58	76.67	75.95	0.76	0:13
15	Ensemble Stacking (Naïve Bayes (kernel), Random Forest)	83.30	78.95	76.89	0.78	0:06
16	Ensemble Stacking (Voting, Random Forest)	84.78	76.94	78.41	0.78	0:22
17	Ensemble Stacking (Voting, Bagging)	**85.60**	79.14	78.98	**0.79**	1:15

6. Conclusion and future work

Cyber threat intelligence provides security analyst with the information about previous incidents that can help them protect their networks by mitigating all those weak points which have been exploited by the attackers. There are several standards available for representing cyber threat information. Also, there are several open source CTI repositories available. The mass of CTI data is increasing with a very fast pace. It becomes difficult for the security analyst to analyze the massive CTI information and apply mitigations in real time. In this paper, a mechanism to visually analyze CTI information based on machine learning techniques is presented. The solution given here enables security analyst to dig out interesting patterns from CTI and perform analysis from multiple perspectives. The security analyst is capable of identifying frequent TTP, domains, IP addresses, and file formats of past cyber incidents. The country-wise frequency of TTP, and IP addresses is also provided. The time duration in which a particular cyber incident took place is also reported in the visual analysis. The TTP dataset is used to train several machine learning models. The trained models are predicting cyber threat actors and malware with a high accuracy of 86%. In future, more cyber threat repositories will be analyzed along with more effective visual analytics. An open source web application will be developed that is capable of taking as input any CTI standard and provide effective visual analytics. It will also be possible to predict unseen cyber threat actors based on their TTP.

References

1. Hackmageddon: June 2021 Cyber Attack Statistics.
   https://www.hackmageddon.com/category/security/cyber-attacks-statistics/
2. Bartoli, A., A. de Lorenzo, E. Medvet, M. Faraguna, F. Tarl. A Security-Oriented Analysis of Web Inclusions in the Italian Public Administration. – Cybernetics and Information Technologies, Vol. **18**, 2018, No 4, pp. 94-110.
3. US-CERT: United States Computer Emergency Readiness Team.
   https://www.us-cert.gov/tlp
4. OpenIOC: An Open Framework for Sharing Threat Intelligence.
   http://www.openioc.org/
5. VERIS: The Vocabulary for Event Recording and Incident Sharing.
   http://veriscommunity.net/
6. IODEF Design principles and IODEF Data Model Overview.
   https://www.terena.org/activities/tf-csirt/meeting5/demchenko-iodef-design-datamodel.pdf
7. Cyber Observable eXpression: A Structured Language for Cyber Observables.
   https://cybox.mitre.org/
8. Structured Threat Information eXpression: A Structured Language for Cyber Threat Intelligence Information.
   http://stix.mitre.org/
9. Trusted Automated eXchange of Indicator Information: Enabling Cyber Threat Information Exchange.
   http://taxii.mitre.org/
10. Ten of the Best Threat Intelligence Feeds.
    https://d3security.com/blog/10-of-the-best-open-source-threat-intelligence-feeds/

11. Hail-a-Taxii.
http://hailataxii.com/
12. ATT&CK MITRE.
https://attack.mitre.org/
13. V e n k a t r a m, K., G. A. M a r y. Review on Big Data & Analytics – Concepts, Philosophy, Process and Applications. – Cybernetics and Information Technologies, Vol. **17**, 2017, No 2, pp. 3-27.
14. Stixproject.github.io. (2019). About STIX | STIX Project Documentation.
https://stixproject.github.io/about/
15. S t r o m, B. E., A. A p p l e b a u m, D. P. M i l l e r, K. C. N i c k e l s, A. G. P e n n i n g t o n, C. B. T h o m a s. Mitre att&ck: Design and Philosophy. Technical Report, 2018.
16. IBM X Force Exchange.
https://exchange.xforce.ibmcloud.com/
17. Symantec Cyber Security.
https://www.broadcom.com/products/cyber-security
18. Cyber Security Experts and Solution Provider.
https://www.fireeye.com/
19. CrowdStrike: Leader in Endpoint Protection.
https://www.crowdstrike.com/
20. B r o m i l e y, M. Threat Intelligence: What It Is, And How to Use It Effectively. – SANS Institute InfoSec Reading Room, Vol. **15**, 2016, 172.
21. C r a i g, M., A. L a k h o t i a, C. L e D o u x, A. N e w s o m, V. N o t a n i. VirusBattle: State-of-the-Art Malware Analysis for Better Cyber Threat Intelligence. – In: Proc. of 7th International Symposium on Resilient Control Systems (ISRCS'14), IEEE, 2014, pp. 1-6.
22. STIXViz. (n.d.). Utilities & Developer Resources.
http://stixproject.github.io/documentation/utilities/
23. N o e l, S. Interactive Visualization and Text Mining for the Capec Cyber Attack Catalog. – In: Proc. of ACM Intelligent User Interfaces Workshop on Visual Text Analytics, 2015, pp. 1-8.
24. Zoomable Sunburst.
https://bl.ocks.org/mbostock/4348373
25. Pebbles – Using Circular Treemaps to Visualize Disk Usage.
http://lip.sourceforge.net/ctreemap.html.
26. FoamTree: Interactive Voronoi Treemap (n.d.).
https://carrotsearch.com/foamtree
27. Z h a o, H., L. L u. Variational Circular Treemaps for Interactive Visualization of Hierarchical Data. – In: Proc. of IEEE Pacific Visualization Symposium (PacificVis'15), IEEE, 2015. pp. 81-85.
28. D a n i e l, B., M., A. E n d e r t, D. K i d w e l l. 7 Key Challenges for Visualization in Cyber Network Defense. – In: Proc. of 11th Workshop on Visualization for Cyber Security, 2014, pp. 33-40.
29. C a w t h o n, N., A. V. M o e r e. The Effect of Aesthetic on the Usability of Data Visualization. – In: Proc. of 11th International Conference Information Visualization (IV'07), IEEE, 2007, pp. 637-648.
30. B r o n w y n, W., S. J. P e r l, B. L i n d a u e r. Data Mining for Efficient Collaborative Information Discovery. – In: Proc. of 2nd ACM Workshop on Information Sharing and Collaborative Security, 2015, pp. 3-12.
31. S i n g h, N., S. S. K h u r m i. Malware Analysis, Clustering and Classification: A Literature Review. – Int. J. Comput. Sci. Technol., Vol. **6**, 2015, No 1, pp. 68-72.
32. Z a h r a, B., H. H a s h e m i, S. M. H. F a r d, A. H a m z e h. A Survey on Heuristic Malware Detection Techniques. – In: Proc. of 5th Conference on Information and Knowledge Technology, IEEE, 2013, pp. 113-120.
33. K y l e, O'M., D. S h i c k, J. S p r i n g, E. S t o n e r. Malware Capability Development Patterns Respond to Defenses: Two Case Studies. White Paper, Software Engineering Institute, Carnegie Mellon University, 2016.
34. S a e e d, I. A., A. S e l a m a t, A. M. A b u a g o u b. A Survey on Malware and Malware Detection Systems. – International Journal of Computer Applications, Vol. **67**, 2013, No 16.
35. A b e d e l a z i z, M., O. A l r a w i. Unveiling Zeus: Automated Classification of Malware Samples. – In: Proc. of 22nd International Conference on World Wide Web, 2013, pp. 829-832.

36. H a n, J., M. K a m b e r. Data Mining. Concepts and Techniques. – In: Morgan Kaufmann. Vol. **340**. 2012. 744 p.
37. I k r a m, S. T., A. K. C h e r u k u r i, B. P o o r v a, P. S. U s h a s r e e, Y. Z h a n g, X. L i u, G. L i. Anomaly Detection Using XGBoost Ensemble of Deep Neural Network Models. – Cybernetics and Information Technologies, Vol. **21**, 2021, No 3, pp. 175-188.
38. ANOMAL STAXX.
https://www.anomali.com/resources/staxx
39. N o o r, U., Z. A n w a r, A. W. M a l i k, S. K h a n, S. S a l e e m. A Machine Learning Framework for Investigating Data Breaches Based on Semantic Analysis of Adversary's Attack Patterns in Threat Intelligence Repositories. – Future Generation Computer Systems, Vol. **95**, 2019, pp. 467-487.
40. UmaraNoor/CTI-Visualizations-Using-R.
https://github.com/UmaraNoor/CTI-Visualizations-Using-R-

Received: 09.09.2021; Second Version: 17.01.2022; Accepted: 20.04.2022