THREAT ACTION EXTRACTION USING INFORMATION RETRIEVAL

Chia-Mei Chen[1], Jing-Yun Kan[1], Ya-Hui Ou[2],
Zheng-Xun Cai[1] and Albert Guan[3]

[1]Department of Information Management,
National Sun Yat-sen University, Taiwan
[2]National Penghu University of Science and Technology, Taiwan
[3]Department of Applied Mathematics, National Sun Yat-sen University, Taiwan

ABSTRACT

To gain insight into potential cyber threats, this research proposes a novel automatic threat action retrieval system, which collects and analyzes various data sources including security news, incident analysis reports, and darknet hacker forums and develops an improved data preprocessing method to reduce feature dimension and a novel query match algorithm to capture effective threat actions automatically without manually predefined ontology applied by the past research. The experimental results illustrate that The proposed method achieves an accuracy of 94.7% and a recall rate of 95.8% and outperforms the previous research. The proposed solution can extract effective threat actions automatically and efficiently.

KEYWORDS

cyber threat intelligence, word vector, information retrieval.

1. INTRODUCTION

Organizations and businesses apply modern information technologies to expand services and improve customer satisfaction, while in the meantime they are facing potential cyberattacks. Cyberattacks have increased in frequency and sophistication, presenting significant challenges for organizations that must defend their data and systems from capable threat attackers. They utilize a variety of tactics, techniques, and procedures (TTPs) to compromise systems, disrupt services, commit financial fraud, and expose or steal intellectual property and other sensitive information. Given the risks these threats present, organizations seek solutions to improve information security and reduce cyberattack risks.

TTPs are the patterns of activities or methods associated with a specific threat actor or group of threat actors [1], which help to identify common attack vectors and possible vulnerable systems likely compromised. Among the key elements of TTP information, identify threat actions is the most essential for understanding TTPs and proactively defending against cyberattacks.

Machine learning techniques have been applied to CTI research recently. Most past research focused on classifying security and non-security related documents or extracting vulnerabilities [2-5] but rarely extracting attack tactics to fill up the information needed by APT incidents to outline attack processes. Some previous work [6-8] manually built up a TTP ontology that consumes intensive labor work and requires to keep it updated as new attack vectors emerge.

David C. Wyld et al. (Eds): CCSIT, SIPP, PDCTA, AISC, NLPCL, BIGML, NCWMC - 2021
pp. 13-19, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.110702

To obtain efficient threat actions, such as hide malicious operations, avoid raising suspicion, and contain .scr file, cybersecurity staff needs to acquire a wide range of articles in order to comprehend the information. Based on the reading speed statistics from ExecuRead [9], the reading speed of technical articles is 50~75 wpm, which takes 5 ~ 6 minutes per page. For a mid-size APT report [10] of 12 pages, a reader needs one hour or so to comprehend the threat actions in the report and may miss some. Such a task is labor-intensive and desires an efficient and automatic threat action retrieval method.

To our best knowledge, the present study is the first attempt to automatically identify threat actions without manually defined ontology by applying multiple word vector models. This research proposes a CTI retrieval method that extracts a key threat action list, which replaces the role of ontology applied by the previous research. Furthermore, the proposed method develops a new query match algorithm that combines multiple word vector language models and similarity functions to capture effective threat actions automatically.

The primary contribution of this study is discovering potential cybersecurity information by exploring multiple types of data sources and multiple state-of-the-art word vector models and developing a novel information retrieval method that extracts threat actions automatically without ontology.

The remainder of the paper is structured as follows. Section 2 reviews the state of the art in the scope of threat intelligence extraction and natural language processing approaches. Section 3 presents the proposed threat action extraction method, followed by the performance evaluation and discussion in Section 4. The last section draws the conclusion remark and the future directions of this study.

2. LITERATURE REVIEW

Settanni et al. [11] evaluated their proposed document correlation methods, where a document is represented as a feature vector, and demonstrated that features based on TF-IDF from the document's own words perform better and those from pre-determined dictionary exhibit a low accuracy and precision.

Niakanlahiji et al. [12] employed a context-free grammar (CFG) model to extract candidate threat actions and applied TF-IDF to extract threat actions. Their results imply that TF-IDF is suitable for representing the importance of a candidate threat action among a list of tokens, so this study adopts it for extracting relevant short phrases from candidate threat actions.

Distributed representations of words in a vector space help learning algorithms to achieve better performance in NLP tasks by grouping similar words. Word2Vec (W2V) [13] is a family of word embedding (word vector) models of representing distributed representations of words in a corpus, where Continuous Bag-of-Words Model (CBOW) and Continuous Skip-gram Model are commonly used. Word2Vec is a two-layer neural network and produces a vector space, where each unique word in a corpus is assigned a corresponding vector in the space.

A study [14] concluded that Word2Vec outperforms the traditional feature selection models including CHI, IG, and DF. As words may have different meanings (i.e., senses) depending on the context, identifying words in the correct meaning is important for extracting relevant information. Two previous studies [15, 16] concluded that Cosine similarity and Word2Vec can effectively capture syntactic word similarities and outperforms LSA (Latent semantic analysis) commonly used in word sense disambiguation. Both applied WordNet [17] as the evaluation

corpus. WordNet is a large lexical database of English, where nouns, verbs, adjectives, and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept.

Word2Vec models lose the ordering of the words and ignore the semantics of the words. An unsupervised algorithm Doc2Vec (D2V) [18] represents each document by a dense vector, which overcomes the weaknesses of Word2Vec. Kadoguchi et al. [19] applied Doc2Vec and ML technology to classify information security data from dark web forums, and the results indicate that Doc2Vec is effective on feature selection and a multi-layer classifier can achieve 79% accuracy. Another study [20] applied Doc2Vec with Cosine similarity on classifying court cases and yields 80% accuracy. A performance study [20] demonstrated that Word2Vec and Doc2Vec perform better than N-gram on text classification and semantic similarity.

If a word is not in the training corpus, Word2Vec fails to identify its similar words. FastText [21, 22] improves the drawback of Word2Vec by applying N-gram to build on not just using the words in the training vocabulary but also their substrings. FastText became popular and replaced Word2Vec on text classification [23, 24] after it was invented. A study demonstrated that FastText achieves 78% accuracy better than Word2Vec and Doc2Vec on text classification; another study [25] drew a similar conclusion remark.

TTPDrill [6] adopted Stanford typed dependency parser to extract candidate threat actions and then mapped these candidate threat actions to those in a pre-defined ontology based on BM25 [26] similarity score. A follow-up study, ActionMiner, [8] improved the above parser of candidate action extraction by applying entropy and mutual information to understand the specificity of verbs used in cybersecurity reports. A study [7] manually selected threat actions, classified the features of the selected threat actions, and associated the threat actions and malware by random forest. The above research all depend on an ontology defined manually and labor-intensive. This study proposes an automatic method to construct a key threat action list in replacement of a manually defined ontology and employs ML technology to analyze and identify effective threat actions.

3. THE PROPOSED METHOD

Figure 1 overviews the major components of the proposed CTI retrieval method. In the model building, it retrieves documents from the security-related websites and sanitizes the text content, and then labels all the tokens by applying the part-of-speech tagging method, extracts verb-form tokens as a candidate threat action list.

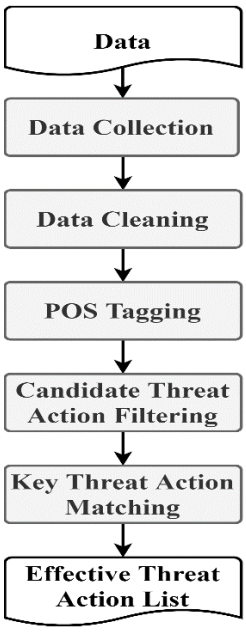

Figure 1. The architecture of the proposed method (source from this study).

Common NLP techniques are adopted to reduce the feature dimension, including tokenization, stop word removal, stemming, and lemmatization. Data cleaning reduces the word density in a given text and helps in preparing the accurate features for model training, as cleaned data improves the efficiency of ML models. The candidate threat actions are extracted from the above-cleaned text by applying POS tagging.

As an ontology requires intensive labor work and its efficiency heavily relies on the completeness of the human-defined ontology, the proposed method plans to automatically build up a key threat action list that is the key component of a TTP ontology to reduce the dependency on domain knowledge. To construct a key threat action list, a two-stage process is developed: the first stage filters out non-security related action tokens and the second stage applies similarity matching measure to extract key threat actions, where the key threat action list serves the purpose of the ontology used in the past research in threat action retrieval.

4. SYSTEM VALIDATION AND EVALUATION

The dataset is sourced from information security reports, Github's APTNotes [27], where APTNotes have acquired comprehensive APT attack investigation reports published by cybersecurity companies, which explain attack chains and threat actions in detail. A total of 600 articles published from 2008 to May 2020 has been collected, where 520 reports from 2008 to 2019 are used for training and the newer ones are for evaluation purpose in order to evaluate if the proposed system can identify threat actions effectively based on a past dataset.

The articles in the dataset are studied, and the threat actions of each article are labeled by a security professional for performance evaluation, where the labeled dataset is summarized in Table 1.

Table 1. The threat actions of the dataset (source from this study).

Dataset	No. of threat actions			No. of articles for testing/training
	Candidate	Effective	Invalid	
APTNotes	21,631	1,438	20,193	95/520

To validate the proposed method by comparing compare the performance of different combinations of filtering and similarity methods as listed in Table 2. The experimental results demonstrate that the proposed threat action retrievalmethod achieves the best performance among all the different combinations of filtering and similarity methods and can identify threat actions effectively.

Based on our preliminary study, a security-related article might contain non-security action words that are not related to threat actions. An ontology-based approach requires intensive manual work. Therefore, the study proposes a multi-stage threat action retrieval approach in order to minic the effort of ontology. The data clean removes noise and consolidates synonyms; the POS tagging labels and filters out non-verb tokens; the filtering process removes non-security action verbs; the matching process computes the similarity of the threat actions to retrieve key threat actions. The correctness of the experimental results are validated by humans, and the results verify the study objective: automatically retrieving threat actions without ontology.

Table 2. The performance of the different filtering and matching methods (source from this study).

Filtering	Matching	Precision	Recall	F1-score
TF-IDF	BM25	71.63%	28.55%	40.83%
WordNet	BM25	62.78%	74.80%	68.00%
WordNet	BM25, W2V, D2V, FastText	90.63%	94.55%	92.58%

5. CONCLUSION

This study applies word vector, tagging, filtering techniques to capture threat actions. The novelty of the proposed solution includes automatically producing a key threat action list as the base of the ontology, the two-stage key threat action extraction algorithm, and applying word vector models for key threat extraction.

The experimental results demonstrate that the proposed solution can capture effective threat actions efficiently with high accuracy and outperforms the previous research. According to the results, the proposed method achieves the following research goals: to identify threat actions efficiently without a predefined ontology and to be able to extract threat actions from different types of documents and in different languages.

This study applies part-of-speech tagging to label tokens in a sentence with their grammatical word categories, but it does not maintain grammatical relations between them. The future work might be able to explore dependency parsing to analyze the sentence structure as it keeps tokens' grammatical relations.

As this research focuses on retrieving threat actions, other pieces of CTI information might be useful for attack prevention. Exploring the relationships among adversaries, victims, and threat actions is another possible research direction for understanding the correlations of these parties.

REFERENCES

[1] J. Friedman and M. Bouchard. "Definitive Guide to Cyber Threat Intelligence." https://cryptome.org/2015/09/cti-guide.pdf (accessed: Nov. 11, 2020).

[2] C. Sabottke, O. Suciu, and T. Dumitraș, "Vulnerability disclosure in the age of social media: Exploiting twitter for predicting real-world exploits," in 24th Security Symposium (Security 15), 2015, pp. 1041-1056.

[3] A. S. Gautam, Y. Gahlot, and P. Kamat, "Hacker Forum Exploit and Classification for Proactive Cyber Threat Intelligence," in International Conference on Inventive Computation Technologies, 2019: Springer, pp. 279-285.

[4] L.-J. Wei, "Distinguishing between Intelligence Articles and Technical Articles based on Extracted Keywords and IOC Elements," Master, Dept. of Computer Science, National Taiwan University of Science and Technology, 2017.

[5] N. Dionísio, F. Alves, P. M. Ferreira, and A. Bessani, "Cyberthreat detection from twitter using deep neural networks," in 2019 International Joint Conference on Neural Networks (IJCNN), 2019: IEEE, pp. 1-8.

[6] G. Husari, E. Al-Shaer, M. Ahmed, B. Chu, and X. Niu, "TTPDrill: Automatic and Accurate Extraction of Threat Actions from Unstructured Text of CTI Sources," presented at the Proceedings of the 33rd Annual Computer Security Applications Conference, 2017.

[7] Z. Zhu and T. Dumitraș, "FeatureSmith: Automatically Engineering Features for Malware Detection by Mining the Security Literature," presented at the Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016.

[8] G. Husari, X. Niu, B. Chu, and E. Al-Shaer, "Using entropy and mutual information to extract threat actions from cyber threat intelligence," in 2018 IEEE International Conference on Intelligence and Security Informatics (ISI), 2018: IEEE, pp. 1-6.

[9] ExecuRead. "Speed Reading Facts." https://secure.execuread.com/facts/ (accessed: July. 3, 2020).

[10] YOROI. "The North Korean Kimsuky APT keeps threatening South Korea evolving its TTPs " https://yoroi.company/research/the-north-korean-kimsuky-apt-keeps-threatening-south-korea-evolving-its-ttps/ (accessed: Aug. 3, 2020).

[11] G. Settanni, Y. Shovgenya, F. Skopik, R. Graf, M. Wurzenberger, and R. Fiedler, "Acquiring cyber threat intelligence through security information correlation," in 2017 3rd IEEE International Conference on Cybernetics (CYBCONF), 2017: IEEE, pp. 1-7.

[12] S. Chandel, J. Wei, and B.-T. Chu, "A natural language processing based trend analysis of advanced persistent threat techniques," in 2018 IEEE International Conference on Big Data (Big Data), 2018: IEEE, pp. 2995-3000.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Advances in neural information processing systems, 2013, pp. 3111-3119.

[14] W. Tian, J. Li, and H. Li, "A method of feature selection based on Word2Vec in text categorization," in 2018 37th Chinese Control Conference (CCC), 2018: IEEE, pp. 9452-9455.

[15] K. Orkphol and W. J. F. I. Yang, "Word sense disambiguation using cosine similarity collaborates with Word2vec and WordNet," vol. 11, no. 5, p. 114, 2019.

[16] A. Handler, "An empirical study of semantic similarity in WordNet and Word2Vec," 2014.

[17] Princeton University. "WordNet." https://wordnet.princeton.edu (accessed: Nov. 11, 2020).

[18] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in International conference on machine learning, 2014, pp. 1188-1196.

[19] M. Kadoguchi, S. Hayashi, M. Hashimoto, and A. Otsuka, "Exploring the Dark Web for Cyber Threat Intelligence using Machine Learning," in 2019 IEEE International Conference on Intelligence and Security Informatics (ISI), 2019: IEEE, pp. 200-202.

[20] L. T. B. Ranera, G. A. Solano, and N. Oco, "Retrieval of Semantically Similar Philippine Supreme Court Case Decisions using Doc2Vec," in 2019 International Symposium on Multimedia and Communication Technology (ISMAC), 2019: IEEE, pp. 1-6.

[21] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," Transactions of the Association for Computational Linguistics, vol. 5, pp. 135-146, 2017.

[22] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," arXiv preprint arXiv:1607.01759, 2016.

[23] V. Zolotov and D. J. a. p. a. Kung, "Analysis and optimization of fasttext linear text classifier," 2017.
[24] I. Santos, N. Nedjah, and L. de Macedo Mourelle, "Sentiment analysis using convolutional neural network with fastText embeddings," in 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI), 2017: IEEE, pp. 1-5.
[25] D. Gromann and T. Declerck, "Comparing pretrained multilingual word embeddings on an ontology alignment task," in Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018.
[26] S. Robertson and H. Zaragoza, The probabilistic relevance framework: BM25 and beyond. Now Publishers Inc, 2009.
[27] kbandla. "Aptnotes." https://github.com/aptnotes/data (accessed: Nov. 11, 2020).