Retrieving Potential Cybersecurity Information from Hacker Forums

Chia-Mei Chen[1], Dan-Wei Wen[2], Ya-Hui Ou[3], Wei-Chih Chao[1], and Zheng-Xun Cai[1]
(Corresponding author: Chia-Mei Chen)
Department of Information Management, National Sun Yat-sen University[1]
No. 70, Lianhai Rd, Gushan District, Kaohsiung 804, Taiwan
(Email: cchen@mail.nsysu.edu.tw)
Department of Management Sciences, Tamkang University, Taipei, Taiwan[2]
National Penghu University of Science and Technology, Taiwan[3]
(Received Mar. 6, 2021; Revised and Accepted Sept. 21, 2021; First Online Oct. 19, 2021)
Special Issue on Multimedia Application on Information Hiding Forensics and Cybersecurity

Abstract

To adapt to the rapidly evolving cyberattacks, cyber threat knowledge is essential for organizations to gain visibility into the fast-evolving threat landscape and timely identify early signs of an attack and the adversary's strategies, tactics, and techniques. In addition, to gaining insight into potential cyber threats, hacker forums are a valuable source. However, the complexity and diversity of the content in hacker forums make it challenging to retrieve useful cybersecurity information. This research proposes an improved data preprocessing method to reduce feature dimension and a hybrid method combining text tagging and clustering analysis techniques to discover cybersecurity information from unstructured hacker forums. The experimental results illustrate that the proposed solution could extract cybersecurity information efficiently.

Keywords: Cyber Threat Intelligence, Hacker Forum, Latent Dirichlet allocation, Natural Language Processing

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.

According to a guide to cyber threat information sharing published by the National Institute of Standards and Technology (NIST) [16], cyber threat information or cyber threat intelligence (CTI) is any information that can help an organization identify, assess, monitor, and respond to cyber threats. Cyber threat information includes indicators of compromise (IoC); tactics, techniques, and procedures used by threat actors; suggested actions to detect, contain, or prevent attacks; and the findings from the analyses of incidents. Organizations can improve their security postures in case such cybersecurity information is acquired.

Collecting such cybersecurity information is an important investment for organizations as it provides a proactive measure to prevent security breaches and saves financial losses. To obtain CTI, security teams gather unstructured data from multiple sources and analyze it to retrieve useful CTI about adversaries and attack signatures to make security decisions for organizations. The purpose of such CTI collection and discovery is to keep organizations informed of the potential threats and exploits.

Hacker forums are a popular internet community for hackers sharing hacking knowledge such as security breaches, hacking tools, malware, evasion techniques, and data leakage. For example, hackers discussed attack plans in the forums [44]; 7.5 million customer personal information was leaked from an online financial service company and sold in hacker forums [7]; a data breach broker sold databases of user records from 14 companies [37]; some forums offer hackers hiring, penetration test, and remote access services [29].

Hacker forums are a valuable source of cybersecurity intelligence [15]. Due to the massive volume of forum posts, extracting cybersecurity-related information from hacker forums is important to discover potential threats and security trends. Therefore, this study extracts infor-

mation from hacker forums to discover vital cyber threat information to facilitate prompt response to cyberattacks.

Classification is a supervised learning approach that learns to figure out what class a new object should fit in by learning from training data with the class labels; clustering is an unsupervised learning approach that groups similar objects without knowing what their labels are. Classification uses predefined classes in which objects are assigned, while clustering identifies similarities between objects, which it groups according to those features in common and which differentiate objects from other groups. Therefore, classification could be used to detect patterns such as IoC (Indicator of Compromise) patterns, malicious URLs, domain names, etc.; clustering could be used to explore forum content and discover new information discussed in the forums. To identify threat intelligence, most literature applied either classification or cluster models [1, 2, 9]. This study combines the two approaches, text tagging and clustering, to explore the content of hacker forums and to discover the CTI information.

One key challenge of clustering is how to determine the number of clusters, as it depends on the level of granularity and analysis goals. This study compares different clustering models with various clustering evaluation measures including the elbow method, Silhouette Coefficient, Calinski-Harabaz Index, and Davies-Bouldin Index to find a valid approach to determine the number of clusters and discover CTI in hacker forums.

Hacker forums are supposed to discuss and share hacking-related subjects, while users may post freestyle or random information. Such posts would make analysis and extraction complicated. To propose an effective CTI extraction method for hacker forums, this study improves the traditional data cleaning method and reduces the feature dimension greatly. The posts in hacker forums contain diverse technical as well as non-technical related information. Therefore, the study proposes a novel analysis method that adopts two-stage clustering to identify new threat information, where the first stage clustering groups the content by theme topics and the second stage focuses on dividing into security-related event clusters.

2 Research Gaps And Questions

Several research gaps were identified from the literature review. First, current CTI efforts rely on the use of autofeeds from security vendors to generate threat intelligence. This means current security measures are often handled reactively based on existing attack cases. Second, hacker forums contain diverse non-security related information and free-style writing forms, which require effective data cleaning and clustering to extract security-relevant information. Finally, previous work focused on identifying security information by classification with patterns and rarely explored forum content to discover potential threat intelligence by clustering. With these research gaps, the following research questions have been proposed to guide the study:

- How to pre-process forum posts effectively to extract meaningful content?
- How to validate the effectiveness of the clustering results?
- How to explore hacker forums and extract proactive CTI efficiently by clustering?

The primary contribution of this study is to discover potential cybersecurity information by exploring hacker forums as a source of cyber threat intelligence and by applying a hybrid method of text tagging and clustering. This is achieved by using an automated process that consists of the following main phases: (1) data collection, (2) data cleaning and tagging, and (3) two-stage clustering of discovering topics pertaining to cybersecurity.

3 Literature Review

From the perspective of data collection, data can be divided into two categories: indicator-based and document-based. The first is indicator-based data feeds (Indicator Feeds). Indicator-based data feeds mainly share indicators of compromise (IoC) to achieve attack prevention in a short time, including the blacklist IP address, malicious domains, and malware hashes. The document-based data may contain rich and comprehensive threat information than the former one, which requires to apply NLP techniques and analysis models to retrieve them.

Tagging is efficient in extracting indicator-based CTI information as well as semantic information from unstructured corpus. Wollschlaeger *et al.* [43] proposed a semantic annotation framework based on tagging, where the tags address several independent aspects of semantics, increasing the expressiveness of information semantics. Wang and Chow [44] performed semantic extraction by tagging unstructured CTI data, and the experiment results show that the extracted entities and relationships by tagging provide valuable CTI information. Chen *et al.* [5] utilized tagging for capturing the semantics of web services in order to improve clustering performance.

The term frequency-inverse document frequency (TF-IDF) is a numerical statistic that reflects the importance of a word to a document in a collection of documents or corpus, where TF refers to the total number of times a given word appears in a document against the total number of all words in the document and IDF measures how common or rare a given word is across all documents. The TF-IDF can be expressed in the following equation.

$$tfidf(t,d) = tf(t,d) \times idf(t) \quad (1)$$

where t is a token or a given word and d is the document. The TF-IDF value increases in proportion to the number of times a given word appears in the document but is offset by the frequency of the word in the corpus to adjust the factor of words that frequently appeared.

Niakanlahiji *et al.* [4] 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) [27] 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. It 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 [40] 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 [14, 31] concluded that Word2Vec can capture syntactic word similarities effectively and outperforms LSA (Latent semantic analysis) used commonly in word sense disambiguation.

Word2Vec models lose the ordering of the words. An unsupervised algorithm Doc2Vec (D2V) [22] represents each document by a dense vector, which overcomes the weaknesses of Word2Vec. Kadoguchi *et al.* [17] 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 [34] applied Doc2Vec on classifying court cases and yields 80% accuracy. A performance study [34] demonstrated that Word2Vec and Doc2Vec perform better than N-gram on text classification and semantic similarity.

The above word embeddings are pre-trained models from co-occurrence statistics, while pre-trained contextual language models, BERT (Bidirectional Encoder Representations from Transformers) [10], generate word embeddings by jointly conditioning on left and right context. BERT-based models have been applied for search queries and classifications. Some studies [6, 30, 32] applied BERT for ranking query and document pairs and constructing a search query model, and some [12, 26, 41, 45] utilized BERT-based transformers to detect fake news.

Zhan *et al.* [46] conducted a performance analysis of BERT model and found out that BERT dumps redundant attention weights on tokens with high document frequency, such as periods, and that may lead to a potential threat to the model robustness. BERT extracts representations for query and document in the beginning and relies heavily on the interactions to predict relevance. The authors suggested some improvement may transform it into a more efficient ranking model. Khattab and Zaharia [18] developed an improved BERT-based ranking model that independently encodes the query and the document by delaying interactions. According to the literature review, it might not be suitable for exploring cyber threat information from unlabeled corpus like hacker forums.

Liao *et al.* [25] presented an automatic IoC extraction method based on the observation that the IoCs are described in a predictable way: being connected to a set of terms like "download". It generated 900K IoC items with a precision of 95% and a coverage of over 90%. Kurogome *et al.* [21] proposed an automatic malware signature generation system from given malware samples, and the evaluation demonstrated that the produced IOCs are as interpretable as manually-generated ones.

Samtani *et al.* [36] applied classification and topic modeling techniques to extract source code from manually categorized data, where LDA (Latent Dirichlet allocation) finds the topics of the source code postings and classification categorized the programming language type. Benjamin and Chen [1] utilized recurrent neural network language models (RNNLMs) coupled with methodology from lexical semantics for learning hacker language. They demonstrated that RNNLMs can be used to develop the capability for understanding hacker language and different embedding models may impact the performance of the machine learning model.

Underground forums allow criminals to interact, exchange knowledge, and trade in products and services. Pastrana *et al.* [33] developed a web crawler to capture data from underground forums. Biswas *et al.* [2] applied a logistic regression model and sentiment analysis to achieve role-based hacker classification and examine hacker behaviors in dark forums. The overall classification accuracy is 80.57 %, and the keywords used in message posts are greatly linked to hacker expertise. Gautam *et al.* [11] employed machine learning approaches to classify underground hacker forum data into predefined categories, and the experimental results show that RNN GRU outperforms LSTM and yields the classification results of 99.025% accuracy and 96.56% precision.

Deliu *et al.* [9] explored the potential of Machine Learning (ML) methods to retrieve relevant threat information from hacker forums and compared the text classification performance of a Convolutional Neural Network (CNN) model against a traditional ML approach (Support Vector Machines). They concluded that SVM performs equally well as CNN.

Li *et al.* [23] combined Word2Vec and LDA to cluster academic abstracts and concluded that the combined model clusters the abstracts efficiently. Another study [38] also combined Word2Vec and LDA for web service clustering and demonstrated that the combined model outperforms a plain LDA.

The previous work demonstrated that hacker forums contain valuable CTI and mostly focused on applying classification models for extracting CTI from hacker forums. Traditional ML models can yield high levels of performance that are on par with modern ML models.

4 Methodology

This study developed a CTI discovery method as plotted in Figure 1 to answer the proposed research questions, and the notations used in this study are summarized in Table 1. The proposed method consists of the following components: data collection, data cleaning and tagging, word embedding, and CTI analysis and extraction. This study applies text tagging and word embedding to extract semantic information and develops a two-stage clustering method to retrieve security-related information. According to the literature review, word embedding models could represent semantic information [34], and the studies [5,44] demonstrated tagging could extract useful semantic information and improve clustering performance.

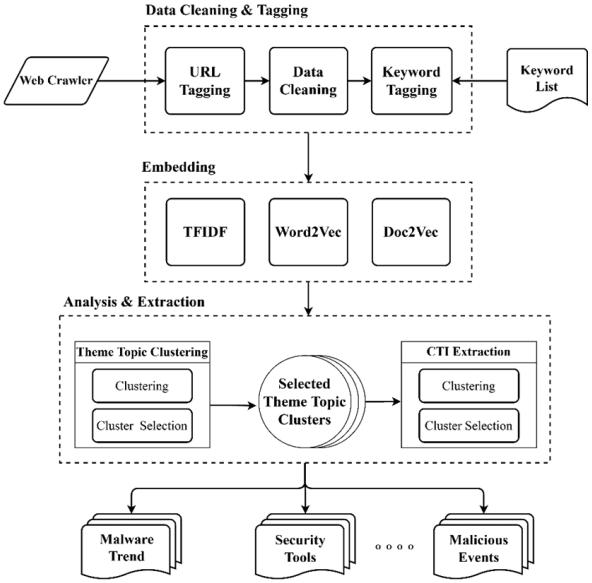

Figure 1: Research design

4.1 Data Collection

Data collection can be achieved by developing a web crawler to gather posts from hacker forums. Some hacker forums employ anti-crawling techniques to hinder automated content extraction, which complicates collection automation. Forum posts may contain various data forms such as text, image, attachment, and threads of responses.

4.2 Data Cleaning and Tagging

The process of data cleaning and tagging reduces the volume of the corpus as well as the dimension of token vectors. This process consists of the following submodules: URL labeling, data cleaning, and keyword tagging, where data cleaning includes tokenization, stop word removal, token pruning, and tagging.

Common data preprocessing in text mining removes URL labels directly before proceeding with the rest of the data preprocessing steps. Li's study [24] observed that sellers might express the privacy information to be sold in a URL-like text format to catch the reader's attention. To retain such information, the proposed method performs URL labeling/tagging before data cleaning, as the text preprocessing steps might remove or disrupt it.

Table 1: Notations used in this study

Notation	Meaning
$\|A\|$	The number of elements in a set A
Corpus	The set of the documents in a corpus
C_{att}	The set of all the theme topic clusters
C_{stt}	The set of the selected theme topic clusters
C_{unfit}	The set of the theme topic clusters in extreme sizes
E_i	The set of the event clusters in the selected theme topic cluster i
TC_i	The theme topic cluster i
W_j	The j-th keyword of a cluster
$W_{TC\_i\_j}$	The j-th keyword in the theme topic cluster i
$S(W)$	The TFIDF score of a keyword W
R_{max}	The maximum ratio of a theme topic cluster to the corpus
R_{min}	The minimum ratio of a theme topic cluster to the corpus
$D(W_j)$	$(S(W_j) - S(W_{j+1}))/S(W_{j+1})$; the discrepancy of the j-th keyword to j+1-th's
H_{dis}	The discrepancy threshold of two consecutive keywords

Users have different writing styles so that the documents often contain different terms with similar meanings. In text mining, a large keyword list (feature set) complicates the analysis and induces bias. Therefore, this study applies text tagging to reduce the feature dimension and to improve the information retrieval performance, while retaining the semantic information. Text tagging is achieved by keyword and regular expression matching in this study. The keyword tagging could achieve the purposes of token pruning and feature dimension reduction. The selected keywords are based on the previous studies [13, 19, 20] and categorized into two types: security and non-security relevant.

The tagged documents contain hashtag tokens in the format of #keyword#, where a matched term or regular expression is replaced by the associated hashtag. Based on our preliminary study on observing posts in hacker forums, tthis study defines 18 hashtags: 7 non-security hashtags (NH) and 11 security hashtags (SH). The non-security hashtags include #HIDDEN#, #IMAGE#, #ATTACHMENT#, #URL#, #QUOTE#, #MODERATOR#, and #PORN#; the security hashtags include #ICQ#, #ACC_PASS#, #E-MAIL#, #WEBSITE#, #EXPLOIT#, #ATTACK#, #MALWARE#, #PROXY#, #PAYMENT#, #TUTORIAL#, #AN-

TIVIRUS#.

The proposed data cleaning process consists of lemmatization and tokenization, stop word removal, irrelevant terms removal by rules. Lemmatization and tokenization divides text information into individual words, where this study deploys word tokenization from Python NLTK as an analysis [28] on open source tools showed that it gives the best output. After the tokenization, noisy text removal steps: punctuation removal, non-ASCII character removal, and stop word removal. A collected English text corpus may contain characters of other languages, such as Chinese, Japanese, or Russian, and such non-English terms are removed to improve the clustering accuracy.

Forum posts normally are not as formal as news articles or technical reports, so they may contain internet slang words, text faces (emoji in the text form), or typos which are non-security related terms for this study. By using the common English words as the base of the stop word list, the proposed data cleaning method acquires more stop words including common internet slang terms [29] to make token pruning more effective. To improve token pruning, it further removes nonsense or non-security terms by regular expression rules such as too long words or with many repeated letters.

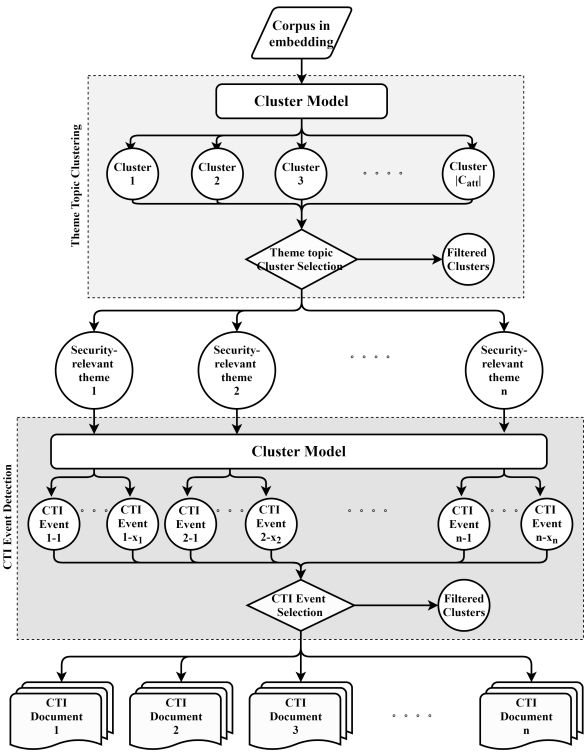

Figure 2: Analysis and extraction process

4.3 Word Embedding

Word embeddings are a type of word representation that allows words with similar meanings to have a similar representation. As the embedding model may affect the proposed clustering performance, this study employs TFIDF to compute term importance and compares two embedding models, Word2Vec and Doc2Vec, in order to find an efficient embedding model.

4.4 Analysis and Extraction

The analysis and extraction module outlined in Figure 2 utilizes a two-stage clustering, where the first clustering (theme topic clustering) determines the theme topics of a corpus and the second clustering (CTI event detection) extracts cyber threat information of each selected topic. As some topic clusters produced from the first clustering may contain non-security related topics or general information without security focus, a set of selection criteria is developed to obtain security-focused topic clusters.

4.4.1 Theme Topic Cluster Selection Criteria

A key issue of cluster analysis is to identify clusters of the subject matter. This study develops a set of selection/filtering rules to extract security-relevant theme topic clusters, where Table 2 outlines the selection criteria. The first two rules exclude the clusters of extreme size, where an extreme size is smaller than the minimum portion or larger than the maximum portion of the corpus and expressed as below.

$$\begin{cases} |TC| < R_{\min} \times \mid \text{Corpus} \mid \\ |TC| > R_{\max} \times \mid \text{Corpus} \mid \end{cases} \quad (2)$$

Table 2: The proposed theme topic cluster selection rules

Rule ID (Action)	Description
TR1 (Removed)	A too-small cluster is removed.
TR2 (Removed)	A loo-large cluster is removed.
TR3 (Selected)	A cluster whose top k keywords are all security hashtags is selected.
TR4 (Removed)	A cluster whose top k keywords could not contribute the most term weighting is removed.
TR5 (Removed)	A cluster whose top m keywords contain non-security hashtags or keywords is removed.

Based on our preliminary study, a large cluster covers a broad range of documents and might not be able to distinguish a specific interest theme, while a small cluster contains little information to form a meaningful theme topic. Based on our preliminary study, a cluster i is considered to be too small, if the number of documents in the cluster is less than 1/50 of the corpus size, i.e.,$|TC_i| < 0.02 \times |Corpus|$;it is too large, if its size is larger than a quarter of the corpus, i.e., $|TC_i| > 0.25 \times |Corpus|$. That is, $R_min = 0.02$ and $R_max = 0.25$. The third rule selects clusters containing security hashtags, which implies that such clusters discuss mostly

security-related information.

Based on our preliminary study by manually examining clustering results, a topic cluster with few keywords of high weighting often contains documents of a specific focus; on the contrary, that with many keywords of similar weighting likely contains diversified documents. Therefore, to identify a focused cluster, the fourth rule checks if there is a large discrepancy drop between two consecutive keywords, It computes the discrepancies of the top k keywords, where the discrepancy of the j-th keyword, $D(W_j) = (S(W_j) - S(W_{j+1}))/S(W_{j+1})$

S(W) is the TFIDF score of a keyword W in the cluster, and j $\in \{1, 2, \ldots, k\}$. If the first k discrepancies are not significant, which implies that this cluster contains no significant focused keywords and is not selected into the list. In this study, k=3, m=10, and a cluster is removed if the discrepancy $D(W_j) < 1.2$. For the fifth rule, a topic cluster containing non-security hashtags or keywords, such as #HIDDEN#, #QUOTE#, thankman, or job, implies that this cluster does not focus on security and is removed from the list.

4.4.2 Determining the Cluster Size

A fundamental step for unsupervised algorithms is to determine the number of clusters into which the data may be clustered. Exploring and retrieving meaningful information efficiently relies heavily on the cluster size. A good clustering produces clusters that are relatively homogeneous within themselves and heterogeneous between each other. Based on this idea, clustering metrics have been proposed to evaluate the quality of clustering results from different aspects. This study selects the number of clusters by considering the following common metrics: elbow method [39], Silhouette Coefficient [35], Calinski-Harabaz Index [3], and Davies-Bouldin Index [8].

4.4.3 CTI Event Detection

After applying the selection rules on the first stage clustering, the proposed system produces a set of security-focused topic clusters. The documents in a single topic cluster contain narrow-domain information as they contain similar keywords. The literature review [23] indicates that clustering narrow-domain texts could be challenging, as narrow-domain leads to keyword overlappings and makes it hard to distinguish sub-domains. As the past research suggests that LDA yields good clustering results, this study employs LDA to perform the second stage clustering. Like the first stage clustering, it may contain non-security focused event clusters, so the following filtering rules are applied.

ER1: A too-small cluster is removed, where a cluster of the size less than 3 is too small.

ER2: A cluster whose top m keywords contain non-security hashtags or keywords is removed.

5 System Evaluation

This study designs the following evaluation to address the proposed research questions as explained below.

- For the first research question, how to preprocess forum posts effectively to extract meaningful content, Experiment I compares the proposed data cleaning method with the traditional approach.
- For the second research question: how to validate the effectiveness of the clustering results, the study defines a clustering effectiveness measure, Embedding Cluster Score (EC_Score), to validate the results of the topic clustering. Experiment II evaluates the efficiency of the proposed method on the topic clustering with different embedding and clustering models.
- For the third research question, how to explore hacker forums and extract proactive CTI efficiently by clustering, this study proposed a hybrid solution that combines text tagging and clustering models to extract CTI information. Experiment III examines the performance of the CTI information extraction.

The study chooses a hacker forum dataset CrackingArena provided by AZSecure to evaluate the proposed solution, which was one of the largest hacker forums existing in 2018 with 11,977 active users. It contains a total of 44,927 posts dated from April 2013 to February 2018.

5.1 Experiment I: Evaluating the Effectiveness of Data Cleaning

Experiment I compares the performance of the proposed data cleaning and tagging method with the traditional data cleaning method that removes common stop words. The resulted corpora after the two data cleaning methods have been validated through human inspection. Table 3 lists the number of posts of each hashtag, and Table 4 lists the number of tokens (word terms) before and after data cleaning and tagging. The results illustrate that the proposed data cleaning and tagging method is effective in reducing the token/feature dimension. The total number of posts is 44,927 and is reduced to 1,543 after the proposed data cleaning process. This experiment also finds out that the forum posts contain quite a lot of nonsense terms such as long words or words with repeated letters.

5.1.1 Performance Measure

To identify an optimal cluster number of a given cluster model, this study considers the following commonly-used clustering metrics: elbow method, Silhouette Coefficient, Calinski-Harabaz Index (CHI), and Davies-Bouldin Index (DBI) as explained in the above section. To compare the performance of the different cluster models, this study defines a performance measure, Embedding Cluster Score (EC_Score), that considers two factors: (1) examining if

Table 3: The number of posts of each hashtag

Hashtag	posts
#HIDDEN#	315
#IMAGE#	791
#ATTACHMENT#	24
#URL#	774
#QUOTE#	171
#MODERATOR#	3
#ICQ#	79
#ACC_PASS#	12
#E-MAIL#	104
#WEBSITE#	113
#EXPLOIT#	30
#ATTACK#	30
#MALWARE#	20
#PROXY#	160
#PAYMENT#	118
#PORN#	99
#TUTORIAL#	55
#ANTIVIRUS#	27

Table 4: The efficiency comparison of token prune

Original token volume	Traditional	This study	
		Without removing nonsense terms	With nonsense terms
50,310	48,909	22,688	20,222

the cluster model can produce security-focused clusters effectively; (2) examining if the cluster model can produce a clustering result of similar-sized clusters.

For the first factor, the effectiveness is examined by the number of the selected theme topic clusters over the total number of the clusters. The selected clusters are security-related, so the more selected clusters imply the cluster model could generate security-focused clusters more effectively.

According to the selection rules listed in Table 2, the extreme-sized clusters are unfitted. For the second factor, a too-large cluster with dense data points implies that the applied word embedding model or the cluster model is not suitable to generate good clustering, while a too-small cluster results from overfitting. Both situations have a negative impact on information retrieval, so the score penalizes them. A good cluster model yields efficient clustering results with security-focused clusters and no or few unfitted clusters. Therefore, the EC_Score is expressed below.

$$EC_score = \frac{|C_{stt}|}{|C_{att}|} \times \left(1 - \frac{|C_{unfit}|}{|C_{att}| - |C_{stt}|}\right) \quad (3)$$

5.2 Experiment II: Evaluating the Performance of Theme Topic Cluster Model

The efficiency of a cluster-based extraction method might depend on with or without word embedding and the applied clustering model. Two embedding models, Word2Vec (W2V) and Doc2Vec (D2V), and their variations are evaluated; three clustering models, K-means, hierarchical cluster (HC), and LDA, are examined. One of the most common approaches, Exp II-1: TFIDF+K-means (without word embedding) is chosen to be the baseline comparison, and a summary of the Exp II results is outlined in Table 5. According to the summarized performance results described in Table 5, Exp II-3: W2V (Skip-Gram)+K-means yields the best theme topic clustering, as it has the highest EC_Score and produces the most security-relevant clusters efficiently without extreme sizes, and Exp II-9 proves to be the worst cluster model. Due to the paper limit, only the clustering results of the baseline, best, and worse clustering models are elaborated in detail, namely Exp II-1 (Baseline): TFIDF + K-means, II-3: W2V (Skip-Gram) + K-means, and II-9: D2V (PV-DM)+ HC.

Table 5: The performance results of Experiment II

EXP ID	$\|C_{att}\|$	$\|C_{stt}\|$	$\|C_{unfit}\|$	EC_Score
Exp II-1(Baseline): TFIDF + K-means	13	5	1	33.7%
Exp II-2: W2V (CBOW) + K-means	19	7	3	27.6%
Exp II-3: W2V (Skip-Gram) + K-means	15	7	0	**46.7%**
Exp II-4: D2V (PV-DM) + K-means	16	3	5	11.5%
Exp II-5: D2V (DBOW) + K-means	17	4	3	18.1%
Exp II-6: TFIDF + HC	16	6	5	18.8%
Exp II-7: W2V (CBOW) + HC	16	5	4	19.9%
Exp II-8: W2V (Skip-Gram) + HC	16	6	2	30%
Exp II-9: D2V (PV-DM)+ HC	26	2	14	3.1%
Exp II-10: D2V (DBOW)+ HC	13	4	4	17.1%
Exp II-11: LDA	11	4	6	5.2%

5.2.1 Exp II-1(Baseline): TFIDF+K-means

Figure 3 shows how to determine the optimal number of clusters by observing the curve changes of the cluster indexes described in the above section, where the navy blue

vertical line indicates an optimal cluster number (13 clusters) and is identified when there are large slope changes appeared in the considered four cluster indexes. Table 6 lists the detailed clustering results and the selected theme topic clusters. The results show that the baseline (TFIDF + K-means) produces a quite good quality of clustering results with only 1 over-sized, unfitted, cluster.

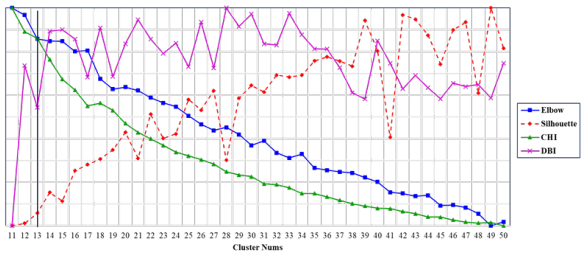

Figure 3: The cluster metrics of Exp II-1: TFIDF+K-means

Table 6: The clustering results of Exp II-1: TFIDF+K-means

ID	Top 3 terms	Rule	Posts
0	#PROXY#, proxy, #URL#'	Selected*	65
1	#IMAGE#, #HIDDEN#, #URL#	TR5	203
2	USER, ACTION, RedURL	Selected	32
3	shell, c99.txt, r57	Selected	23
4	Watchdog, community, stay	TR4	21
5	Proxy, #PROXY#, View	Selected	27
6	#URL#, #PAYMENT#, #IMAGE#	TR2	417
7	#PORN#, Site, #URL#	TR5	61
8	#IMAGE#, #URL#, #QUOTE#	TR5	81
9	account, #IMAGE#, post	TR5	183
10	#URL#, slot, machine	Selected	129
11	site, crack, config	TR5	178
12	FULLZ, Number, GOOD	TR4	36

5.2.2 Exp II-3: Word2Vec(Skip-Gram)+K-means

Figure 4 shows how to determine the optimal number of clusters by observing the curve changes of the cluster indexes described in the above section, where the vertical line indicates an optimal cluster number (15 clusters) and is identified when there are large slope changes appeared in the considered four cluster indexes. Table 7 lists the detailed clustering results and the selected theme topic clusters. The results demonstrate that the combination (W2V(Skip-Gram)+K-means) produces the best quality of clustering among all the cluster and embedding models and no unfitted cluster.

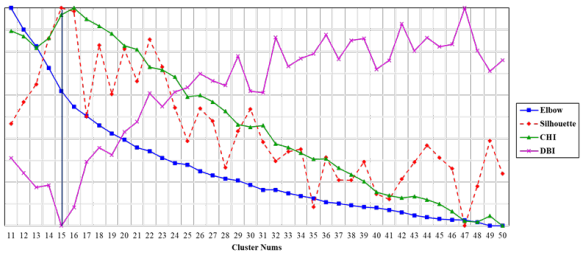

Figure 4: The cluster metrics of Exp II-3: W2V(Skip-Gram)+K-means

Table 7: The clustering results of Exp II-3: W2V(Skip-Gram)+K-means

ID	Top 3 terms	Rule	Posts
0	#URL#, fdfc119f0fb1ddbe54 5829f1777db354	Selected*	50
1	#PROXY#, proxy, list	Selected	58
2	FULLZ, Number, GOOD	TR4	36
3	#IMAGE#, #URL#, site	TR5	294
4	#IMAGE#, account, post	TR5	291
5	#IMAGE#, #URL#, #HIDDEN#	TR5	233
6	USER, ACTION, RedURL	Selected	33
7	shell, c99.txt, r57	Selected	23
8	#IMAGE#, #URL#, #HIDDEN#	TR5	91
9	slot, #URL#, machine	Selected	78
10	Proxy, #PROXY#, View	Selected	28
11	#PAYMENT#, CC, dump	Selected	35
12	stay, community, Watchdog	TR4	25
13	Site, #PORN#, Access	TR5	36
14	#URL#, #IMAGE#, Windows	TR5	145

5.2.3 Exp II-9: Doc2vec (PV-DM) + Hierarchical Cluster

Figure 5 illustrates how to determine the optimal cluster size by observing the curve changes of the cluster in-

dexes described in the above section, where the vertical line indicates an optimal cluster size (26 clusters) is suggested by the indexes. Table 8 lists the detailed clustering results and the selected theme topic clusters. The results show that the combination (D2V(PV-DM)+HC) produces the worst and uneven clustering and could not identify security-focused clusters efficiently, where more than half (14 clusters) are unfitted (TR1 and TR2), about one third (8 clusters) contain non-security related topics (TR5), and only two security-related clusters are selected.

In summary, the results of Exp II indicate that both word embedding and cluster models impact the clustering performance. The worst cluster model fails to distinguish domain-relevant information so that it could not produce efficient clustering results. Furthermore, by comparing the clustering results of the best and worst models (Tables 7 and 8), the number of unfitted clusters affects the clustering efficiency as well, as extreme-sized clusters could not distinguish domain information well.

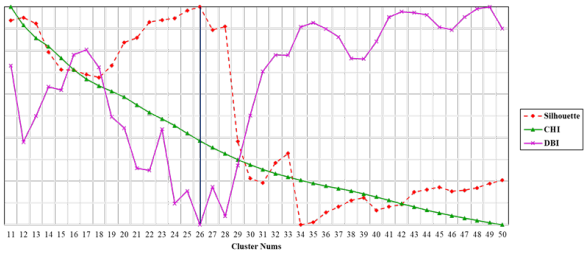

Figure 5: Analysis and extraction process

5.3 Experiment III: Evaluating the Performance of CTI Event Detection Model

If the first stage clustering fails to identify security-focused clusters, the second stage clustering for CTI information extraction might be affected. Therefore, Exp III employs the clustering results from the best cluster model obtained from Exp II (namely, Exp II-3) and adopts LDA to identify CTI events, where Table 9 summarizes the selected clusters from the best cluster model, Table 10 plots the LDA clustering results with coherence validation, and Table 11 outlines the resulted CTI event detection. In Table 10, high coherence indicates the clustering is efficient and could divide the data into a set of meaningful CTI events.

In the theme topic cluster ID: 0, URL Lists, the LDA-based CTI event detection model identifies 2 event clusters: account information and blog lists, where account information includes media platforms like Netflix and RapidGator.Net. The cluster ID: 1, Proxy 1, is further grouped into several types of proxy tools. The cluster ID: 6, System Configuration, contains various system configuration issues including rarefile.net, Sentry, UFC.TV, movies4you.tv, etc., so it is further grouped into 7 clusters. The cluster ID 7, Malicious Script, contains mostly

Table 8: The clustering results of Exp II-9: D2V(PV-DM)+HC

ID	**Top 3 terms**	**Rule**	**Posts**
0	USER, ACTION, GifStart=2	Selected*	29
1	#URL#, #IMAGE#, slot	TR5	182
2	#IMAGE#, #URL#, post	TR5	32
3	#IMAGE#, #URL#, #PORN#	TR5	226
4	CC, Classic, #E-MAIL#	TR1	19
5	#IMAGE#, #URL#, #HIDDEN#	TR5	81
6	#URL#, #IMAGE#, #HIDDEN#	TR5	229
7	#URL#, #IMAGE#, slot	TR5	319
8	#URL#, slot, #IMAGE#	TR5	76
9	der, yang, dan	TR1	5
10	shell, #URL#, c99	TR5	38
11	#URL#, NETFLIX, Site	TR1	9
12	#ACC_PASS#, dump, gold/plat/bus/corp/sign	TR1	5
13	import_module, process_report, process_report_data	TR1	1
14	#URL#, shell, c99.txt	TR4	66
15	#PROXY#, proxy, service	Selected	43
16	FULLZ, Site, GOOD	TR4	68
17	ACTION, recaptcha_response_field= manual_challenge, USER	TR1	3
18	IDM, Internet, download	TR1	2
19	#ACC_PASS#, #ANTIVIRUS#, #URL#	TR1	13
20	#WEBSITE#, DropBox.com, BitShare.com	TR1	1
21	#URL#, /etc/, Apache	TR1	1
22	href, div, /div	TR1	2
23	href, class, /li	TR1	1
24	x15, x78, x75	TR1	1
25	track1/2, -Dumps, pin	TR1	4

malicious php script files shared by the same writer who posted the same script at various times, so it is grouped into one cluster. Likewise, the cluster of Gambling exhibits the same situation and results. The cluster of Proxy 2 is further grouped into two event clusters: proxy code and grabber tools by the LDA cluster model, as both belong to different types of proxy information. The cluster of Dump contains all about credit card information leakage and is further divided into 6 event clusters, where each event cluster contains data leakage from one data breach broker.

By manually examining the LDA clustering results as

Table 9: The selected clusters from the best first stage cluster model (Exp II-3)

ID	Theme topic	Keywords	Posts
0	URL Lists	#URL#, fdfc119f0fb1ddbe545829f1777db354, #E-MAIL#, NETFLIX, #PORN#, MoneyMakingDiscussion.Net, Visit, amateur, March, Bonus	50
1	Proxy 1	#PROXY#, proxy, list, #IMAGE#, combo, Proxy, test, Support, ban, VPN	58
6	System Configuration	USER, ACTION, RedURL, #URL#, blnDigits=1, blnMultiChar=0, Range=0, URLMode=0, Brightness=0, GifOffset=2	33
7	Malicious Script	shell, c99.txt, r57, c99, script, tool, r57.txt, inurl:c100.txt, inurl:c100.php, inurl:locus.txt	23
9	Gambling	slot, #URL#, machine, free, game, casino, Free, play, online, Slot	78
10	Proxy 2	Proxy, #PROXY#, View, Click, Code, #URL#, Text, Attention, directly, Sign	28
11	Dump	#PAYMENT#, CC, dump, #ICQ#, Classic, Dumps, #E-MAIL#, sell, Gold, Canada	35

Table 10: The LDA event clustering results

ID	Theme topic	Event topics	Alpha	Beta	Coherence
0	URL lists	2	0.71	0.11	0.6307
1	Proxy 1	16	0.11	0.21	0.6083
6	System configuration	7	0.61	0.91	0.6076
7	Malicious script	1	0.01	0.01	0.5944
9	Gambling	1	0.61	0.81	0.5923
10	Proxy 2	2	0.21	0.21	0.5831
11	Dump	6	0.81	0.01	0.5819

Table 11: The extracted CTI information

ID	Theme topic	Event cluster	Posts
0	URL lists	Account/password information of media platforms	8
		Russia blog lists	42
1	Proxy 1	Sockshub/rsocks	7
		Fast Proxy Tester/ Checker	11
		ProxyFire	5
6	System configuration	Various system config info	33
7	Malicious script	Sharing php-based malware scripts	23
9	gambling	Tupantitty online gambling	78
10	Proxy 2	Proxy Code	5
		Proxy Grabber	7
11	Dump	Selling privacy data in 6 types	36

described above, the proposed two-stage clustering approach discovers CTI information efficiently. In summary, based on the above three experiments, the evaluation concludes that the proposed CTI information retrieval method can explore hacker forums well and extract cybersecurity information efficiently.

6 Conclusion

Acquiring cyber threat knowledge is essential for organizations to gain visibility into the fast-evolving threat landscape. Hacker forums play an important role in disseminating threat information and correlate significantly with the number of cyber-attacks observed in the real world [42]. Most past research focused on identifying threat intelligence with patterns by classification models. Clustering and preprocessing the content of hacker forums is challenging as the number of clusters is hard to determine and forum writers tend to write freestyle and diversified article posts.

This study applies NLP, tagging, and clustering techniques to explore and capture cybersecurity information in hacker forums. The proposed CTI information retrieval method applies tagging and Word2Vec word embedding to extract key features and employs K-means and LDA two-stage clustering to discover CTI information from unstructured data. Based on Exp I, the proposed data cleaning and tagging method reduces the feature dimension significantly by more than two times better than the traditional data cleaning method, from the size of 48,909 to 20,222. Exp II and III demonstrate that the proposed theme topic cluster selection criteria trim off non-security relevant clusters effectively and the two-stage clustering method can capture cybersecurity-related article posts efficiently.

For determining the clustering size, this study finds out that considering multiple cluster evaluation metrics is effective in finding good clustering parameters. The proposed performance metric, EC_Score, is proved to be helpful for determining the best combination of word embedding and clustering models. This study has demonstrated that applying both text classification and clustering models can achieve great performance in exploring

and extracting CTI information efficiently.

Future work can extend this research to explore online hacker forums in multiple languages or increase understanding of other hacker online community platforms. In addition to increasing the variety of platforms or languages, future work can look at social relationships among hackers and hacker groups or identifying the members creating and disseminating CTI by using social network analysis techniques. This work can also be expanded by introducing a temporal component to track the prevalence of a specific CTI topic over time, which is useful for identifying emerging CTI technologies.

References

[1] V. Benjamin and H. Chen, "Developing understanding of hacker language through the use of lexical semantics," in *IEEE International Conference on Intelligence and Security Informatics (ISI'15)*, pp. 79–84, 2015.

[2] B. Biswas, A. Mukhopadhyay, and G. Gupta, "Leadership in action: How top hackers behave a big-data approach with text-mining and sentiment analysis," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[3] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics-theory and Methods*, vol. 3, no. 1, pp. 1–27, 1974.

[4] S. Chandel, J. Wei, and B. T. Chu, "A natural language processing based trend analysis of advanced persistent threat techniques," in *IEEE International Conference on Big Data (Big Data'18)*, pp. 2995–3000, 2018.

[5] L. Chen, L. Hu, Z. Zheng, J. Wu, J. Yin, Y. Li, and S. Deng, "Wtcluster: Utilizing tags for web services clustering," in *International Conference on Service-Oriented Computing*, pp. 204–218, 2011.

[6] Z. Chen, M. Trabelsi, J. Heflin, Y. Xu, and B. D. Davison, "Table search using a deep contextualized language model," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 589–598, 2020.

[7] CloudSEK Threat Intelligence Team, *Dave Suffers Breach, 7.5m Users' Data Leaked, Meow Attack Deletes 4,000 Unsecured Databases, and More*, Sept. 13, 2020. (`https://cloudsek.com/threatintel/dave-suffers-breach-7-5m-users-data-leaked-meow-attack-deletes-4000-unsecured-databases-and-more/`)

[8] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979.

[9] I. Deliu, C. Leichter, and K. Franke, "Extracting cyber threat intelligence from hacker forums: Support vector machines versus convolutional neural networks," in *IEEE International Conference on Big Data (Big Data'17)*, pp. 3648–3656, 2017.

[10] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[11] 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*, pp. 279–285, 2019.

[12] M. Guderlei, M. Aßenmacher, "Evaluating unsupervised representation learning for detecting stances of fake news," in *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6339–6349, 2020.

[13] A. Gupta and A. Anand, "Ethical hacking and hacking attacks," *International Journal of Engineering and Computer Science*, vol. 6, no. 4, 2017.

[14] A. Handler, *An Empirical Study of Semantic Similarity in WordNet and Word2Vec*, Theses and Dissertations, University of New Orleans, 2014.

[15] InfoSecurity, *Hackers Forums Provide Sense of Community, Information Security Intelligence*, Sept. 13, 2020. (`https://www.infosecurity-magazine.com/news/hackers-forums-provide-sense-of-community/`)

[16] C. Johnson, M. Badger, D. Waltermire, J. Snyder, and C. Skorupka, *Guide to Cyber Threat Information Sharing*, Report, National Institute of Standards and Technology, 2016.

[17] M. Kadoguchi, S. Hayashi, M. Hashimoto, and A. Otsuka, "Exploring the dark web for cyber threat intelligence using machine leaning," in *IEEE International Conference on Intelligence and Security Informatics (ISI'19)*, pp. 200–202, 2019.

[18] O. Khattab and M. Zaharia, "Colbert: Efficient and effective passage search via contextualized late interaction over bert," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 39–48, 2016.

[19] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv: 1408.5882*, 2014.

[20] S. Kumar and D. Agarwal, "Hacking attacks, methods, techniques and their protection measures," *International Journal of Advance Research in Computer Science and Management*, vol. 4, no. 4, pp. 2253–2257, 2018.

[21] Y. Kurogome, Y. Otsuki, Y. Kawakoya, M. Iwamura, S. Hayashi, T. Mori, and K. Sen, "Eiger: Automated ioc generation for accurate and interpretable endpoint malware detection," in *Proceedings of the 35th Annual Computer Security Applications Conference*, pp. 687–701, 2019.

[22] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, pp. 1188–1196, 2019.

[23] C. Li, Y. Lu, J. Wu, Y. Zhang, Z. Xia, T. Wang, D. Yu, X. Chen, P. Liu, and J. Guo, "Lda meets Word2Vec: A novel model for academic abstract clustering," in *Companion Proceedings of the the Web Conference 2018*, pp. 1699–1706, 2018.

[24] W. Li, H. Chen, and J. F. Nunamaker Jr, "Identifying and profiling key sellers in cyber carding community: Azsecure text mining system," *Journal of Management Information Systems*, vol. 33, no. 4, pp. 1059–1086, 2016.

[25] X. Liao, K. Yuan, X. F. Wang, Z. Li, L. Xing, and R. Beyah, "Acing the ioc game: Toward automatic discovery and analysis of open-source cyber threat intelligence," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp. 755–766, 2019.

[26] C. Liu, X. Wu, M. Yu, G. Li, J. Jiang, W. Huang, and X. Lu, "A two-stage model based on bert for short fake news detection," in *International Conference on Knowledge Science, Engineering and Management*, pp. 172–183, 2019.

[27] 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*, pp. 3111–3119, 2013.

[28] V. Mohan, "Text mining: Open source tokenization tools: An analysis," vol. 3, pp. 37–47, 2016.

[29] E. Montalbano, *Experts on Seller Floods Hacker Forum with Data Stolen from 14 Companies*, Sept. 13, 2020. (`https://threatpost.com/threat-actors-introduce-unique-newbie-hacker-forum/157489/`)

[30] R. Nogueira, W. Yang, K. Cho, and J. Lin, "Multi-stage document ranking with bert," *arXiv preprint arXiv: 1910.14424*, 2019.

[31] K. Orkphol and W. Yang, "Word sense disambiguation using cosine similarity collaborates with Word2Vec and WordNet," *Future Internet*, vol. 11, no. 5, p. 114, 2019.

[32] R. Padaki, Z. Dai, and J. Callan, "Rethinking query expansion for bert reranking," in *European Conference on Information Retrieval*, pp. 297–304, 2020.

[33] S. Pastrana, D. R. Thomas, A. Hutchings, and R. Clayton, "Crimebb: Enabling cybercrime research on underground forums at scale," in *Proceedings of the World Wide Web Conference*, pp. 1845–1854, 2018.

[34] L. T. B. Ranera, G. A. Solano, and N. Oco, "Retrieval of semantically similar philippine supreme court case decisions using doc2vec," in *International Symposium on Multimedia and Communication Technology (ISMAC'19)*, pp. 1–6, 2019.

[35] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[36] S. Samtani, R. Chinn, and H. Chen, "Exploring hacker assets in underground forums," in *IEEE international conference on intelligence and security informatics (ISI'15)*, pp. 31–36, 2015.

[37] Security Experts, *Experts on Seller Floods Hacker Forum with Data Stolen from 14 Companies*, Sept. 13, 2020. (`https://www.informationsecuritybuzz.com/expert-comments/experts-on-seller-floods-hacker-forum-with-data stolen-from-14-companies/`)

[38] M. Shi, J. Liu, D. Zhou, M. Tang, and B. Q. Cao, "WE-LDA: A word embeddings augmented LDA model for web services clustering," in *IEEE International Conference on Web Services (ICWS'17)*, 2017.

[39] R. L. Thorndike, "Who belongs in the family?," *Psychometrika*, vol. 18, no. 4, pp. 267–276, 1953.

[40] W. Tian, J. Li, and H. Li, "A method of feature selection based on word2vec in text categorization," in *37th Chinese Control Conference (CCC'18)*, pp. 9452–9455, 2018.

[41] R. Vijjali, P. Potluri, S. Kumar, and S. Teki, "Two stage transformer model for covid-19 fake news detection and fact checking," *arXiv preprint arXiv: 2011.13253*, 2020.

[42] Q. H. Wang, W. T. Yue, and K. L. Hui, "Do hacker forums contribute to security attacks?," in *E-Life: Web-Enabled Convergence of Commerce, Work, and Social Life*, pp. 143–152, 2012.

[43] B. Wollschlaeger, E. Eichenberg, and K. Kabitzsch, "Explain yourself: A semantic annotation framework to facilitate tagging of semantic information in health smart homes," in *HEALTHINF*, pp. 133–144, 2020.

[44] W. T. Yue, Q. H. Wang, and K. L. Hui, "See no evil, hear no evil? dissecting the impact of online hacker forums," *MIS Quarterly*, vol. 43, no. 1, p. 73, 2019.

[45] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi, "Defending against neural fake news," *arXiv preprint arXiv: 1905.12616*, 2019.

[46] J. Zhan, J. Mao, Y. Liu, M. Zhang, and S. Ma, "An analysis of bert in document ranking," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1941–1944, 2020.

Biography

Chia-Mei Chen has joined in the Department of Information Management, National Sun Yat-Sen University since 1996. She was the Section Chef of Network Division and Deputy Director, Office of Library and Information Services in 2009-2011. She had served as a coordinator of TWCERT/CC (Taiwan Computer Emergency Response Team/Coordination Center) during 1998 to 2013 and then as a consultant until 2018. Based on her CSIRT experience, she established TACERT (Taiwan Academic Network Computer Emergency Response Team) in 2009. She

was a Deputy Chair of TWISC@NCKU, a branch of Taiwan Information Security Center during 2017 to 2020. She continues working for the network security society. Her current research interests include anomaly detection, network security, machine learning, text mining, and big data analysis.

Dan-Wei Wen is an assistant professor at the Department of Information Management, Tamkang University. She received her Ph.D. from the Department of Business Administration, National Cheng-Kung University. Her research interests include industry dynamics, catching-up strategy, and data mining.

Ya-Hui Ou received her Ph.D. degree from the Department of Information Management, National Sun Yat-sen University in 2017. She is an assistant professor in the Common Education Teaching Center, National Penghu University of Science and Technology, Penghu, Taiwan. Her research interests include network security and statistical analysis.

Wei-Chih Chao has received his Master's degree from the Department of Information Management, National Sun Yat-sen University. Currently he is a software engineer in an information security institute.

Zheng-Xun Cai received his Master's degree from the National Sun Yat-sen University in 2017 and continues pursuing the PhD degree at the same school. His research focuses on digital forensics, network analysis, and intrusion detection.