IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 7, 2020, accepted May 2, 2020, date of publication May 14, 2020, date of current version June 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2994516

Domain-Oriented Topic Discovery Based on Features Extraction and Topic Clustering

XIAOFENG LU[1], (Member, IEEE), XIAO ZHOU[1], WENTING WANG[2], PIETRO LIO'[3], AND PAN HUI[4], (Fellow, IEEE)

[1]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Electric Power Research Institute, State Grid Shandong Electric Power Company, Jinan 230001, China
[3]Computer Laboratory, University of Cambridge, Cambridge, CB2 1TN, U.K.
[4]Computer Science and Engineering Department, The Hong Kong University of Science and Technology, Hong Kong

Corresponding author: Xiaofeng Lu (luxf@bupt.edu.cn)

This work was supported in part by the Science and Technology Project Funding of State Grid Corporation of China under Grant 520626190067, and in part by the National Natural Science Foundation of China under Grant 61472046.

ABSTRACT Topic detection technology can automatically discover new topics on the Internet. This paper investigates domain-oriented feature extraction methods, and proposes a keyword feature extraction method ITFIDF-LP, a subject word feature extraction method LDA-SLP and a topic clustering model based on vector product similarity. A novel Domain-oriented Topic Discovery based on Features Extraction and Topic Clustering (DTD-FETC) model is proposed to analyze open source web of a domain and identify emerging topics in the domain in real time. This article describes a DTD-FETC system built for cyber security domain. It filters and aggregates web for specical security threat topics such as vulnerability and malware, and helps security staff respond quickly and defends against the emerging cyber threats as early as possible. The recall rate, accuracy and F1 value results of the DTD-FETC method applied to the cyber security dataset are all above 0.99.

INDEX TERMS Topic detection, feature extraction, topic clustering, transfer learning, threat intelligence.

I. INTRODUCTION

With the development of the Internet, people have more and more ways to obtain information from the Internet, such as web pages, microblog, Twitter and so on. A lot of information related to a topic is scattered in different spaces on the Internet, making it more and more difficult for people to easily find the multifaceted information about a topic or event. Faced with a large amount of data on the Internet, without efficient tools, it is difficult for decision makers to obtain information about the latest events or topics, so as to make correct decisions. In this case, Topic Detection and Tracking (TDT) technologies have emerged. TDT technology can discover and correlate information about a topic scattered in different places [1]. TDT can be applied in many fields, such as financial analysis, government governance, network security and so on [2].

In financial domain, TDT technology is used to discover the latest specific topics, such as new policies issued by a country, military conflicts. Financial staff use this kind of information to predict future price trends in stock market and precious metals.

In electronic game domain, some game development companies use TDT technology to discover negative reviews and feedback about their products on game forums [3]. They will take corresponding actions and improve related games, which will attract more gamers.

In cyber security domain, security staff collect and track the latest security news or reports daily to help them improve security defense strategies [4], [5].

Cyber threat intelligence is defined as "the set of data collected, assessed and applied regarding security threats, threat actors, exploits, malware, vulnerabilities and compromise indicators" [6]. Traditional threat intelligence research has mainly explored IOC (Indicators of Compromise) from data sources such as logs, then generated threat intelligence in standard format for different security applications [7], [8]. Liao *et al.* proposed the IACE (innovation solution for fully automated IOC extraction) method for automatically extracting IOC (botnet IPs, etc.) from open-source, secure web

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

pages [9]. Cyber threat intelligence is mainly malicious IP, URL, word hashes, in fact, these are only a small part of the threat intelligence. Moreover, this type of threat intelligence does not provide any background information on the attack (for example, the criminal group behind the criminal act). Therefore, relying on this type of information to analyze cyber attack activity and classify events becomes very difficult.

As an integral part of the new generation of defense systems, threat intelligence continues to develop and improve. In recent years, researchers have mainly used data mining and correlation analysis techniques to study threat intelligence [10]. Researchers automatically identify threat topics from fragmented, open source threat data, helping security analysts respond quickly and protect against emerging cyber threats [5].

Lee *et al.* published the first article that used information from Twitter released by security experts as a data source to mine security threat topics [11]. Their approach used a graph clustering model to mine topic words, and then used the mined security topic words as keywords to mine subsequently relevant topic articles of forums and news websites. They successfully used Twitter as a data source from which security topics were discovered in a timely manner. However, the process of mining data sources like Twitter is tedious, and it is easy to mine topic words that are irrelevant to security interests. The vastness and unfocused nature of Twitter as a data source, combined with limited manpower, can make it easy to miss mining relevant articles and information.

Instead, security reports in cyber security technology blogs and web sites sometimes have a comprehensive description of the attack and are more suitable for security practitioners [5]. The descriptions in such web pages are usually informal descriptions of natural language and require careful analysis to restore relevant attack instructions. Over the years, security analysts have completed these descriptions manually. In this article, we study security topic detection from open source web pages with high information relevance, such as security blogs and security newletters. Security reports or newsletters are produced at a high volume and velocity on the Internet [5].

The challenge, however, comes from the effective gathering of such information, which entails significant burdens for timely analyzing a large amount of data. Recorded Future reportedly uses more than 650,000 open web sources to collect IOC [12]. Because these sources generate a lot of information, manual methods have not been able to efficiently extract content, so new technologies are needed to automatically identify and extract the valuable CTI involved.

The goal of our study is to discover new cyber threat topics in real time from open source webpages. The classification model, such as RNN (Recurrent Neural Network), Transformer and BERT (Bidirectional Encoder Representation from Transformers), is to determine the category of a data based on the existing categories, but because the subject of threat intelligence changes every day, classification methods cannot be used to solve this problem. Therefore, we use the clustering model based on topic detection technology to identify topics in multi-source data in real time to form threat intelligence.

While topic detection technologies are relatively mature, domain-oriented topic detection technology has numerous potential applications. A strategy for identifying open-source cyber threat topics in real time is novel. In this paper, we propose a novel Domain-oriented Topic Discovery based on Features Extraction and Topic Clustering (DTD-FETC) method. DTD-FETC analyzes threat data from open source security news platforms, building on existing general topic detection technology and security domain knowledge. Our DTD-FETC method seeks to, identify both emerging threat topics and event continuation of historical topics in real time.

The main contributions of this paper:

1. This paper proposes a Word2vec model combined with transfer learning is proposed to learn feature word vectors. This addresses the problem that datasets in the field of security intelligence are sparse and word vector models cannot use good semantic information to train the feature word vector.

2. Based on the TF-IDF, the topic model, and the entity recognition model and combined with the knowledge in the field of information security, this paper proposes three feature extraction methods applicable to the information security field.

3. Based on the HAC (Hierarchical Agglomerative Clustering) algorithm and centroid linkage method, this paper proposes an improved centroid linkage method based on vector product similarity to improve the topic clustering model.

II. RELATED WORK

A. GENERAL TOPIC DETECTION AND TRACKING

In recent years, the data from social networks such as Twitter and micro-blog is widely used by researchers in topic detection. Researchers focus on hot topics detection by finding new features and improving topic clustering algorithms.

Huang *et al.* studied the topic detection from microblog with high utility pattern (HUP) clustering and proposed a HUPC framework [13]. A pattern is a collection of several terms. Similar patterns usually express similar semantic information. HUP mining is to find out a group of patterns such that the sum of their utilities is maximized. The HUPC framework consists of three components: top-K HUP mining, HUP clustering and post-processing. A combination of KNN classification and modularity-based partition cluster the HUP set into groups. HUP is new feature in Huang's study and topical words are selected from each pattern cluster. Both the feature of an article and the clustering algorithm are different from our method. And we proposed three methods to extract different features.

Comito *et al.* studied synergies between word embedding and clustering methods, and proposed a Word embedding Clustering (WEC) topic detection method [14]. The key feature of WEC clustering method is the similarity measure method. The similarity measure includes both semantic and

lexical characteristics of the contents. Comito used Word2Vec model to produce word embeddings to gain the word's semantic information. The word vector trained by the Word2Vec model contains the word's semantic information and can reflect the linear relationship between words. We not only use Word2Vec technology to extract the semantic information of words, but also apply transfer learning method to further improve the quality of word semantics in cyber security domain.

Atefeh *et al.* studied event detection technology applied to Twitter [15]. Their article organized discussion of event detection mechanisms according to different event types, detection tasks, detection methods. Haitao Zheng used Twitter as their data source also, and proposed a multi-feature topic detection LTDMF framework to make up for poor performance by single feature topic detection, and then clustered the topic based on the Hierarchical Agglomerative Clustering (HAC) clustering model [16]. The features used in LTDMF are temporal feature, geographic feature,Co-occurrence feature and hashtag feature. The features used in [15], [16] are different from the features we extracted based on the structure of the article. In addition, the clustering algorithm Zheng used is prone to the ''clustering effect'' when clustering topics. This paper improves the clustering algorithm and minimizes the ''clustering effect''.

Some researchers improved the performance of topic detection by improving the clustering algorithm. Ding *et al.* proposed Optimized Affinity Propagation HAC (OAP-HAC) clustering algorithms. They used optimized AP algorithms to find cluster center in micro-learning data [17]. They calculated similarity of two texts through calculating the topic of the corresponding probability distribution. Then, they used the hierarchical agglomerative clustering algorithm to clustering based on their cluster centers. In keyword extraction, Ding considered that keywords have different weights when they are in different positions, such as in the title and detail. When our method performs features extraction, it extracts keyword feature, subject word feature, and domain named entity recognition feature. We propose vector product similarity as the similarity measurement to calculate similarity of two documents.

Manai *et al.* used news data as their data source, made a vector space model to represent the text feature as the input of the clustering model, and then proposed hierarchical density-based spatial clustering with noise (HDBSCAN) model for topic detection [18]. The experimental results proved that the proposed clustering model improves topic detection accuracy. However, Manai's topic detection methods only consider a single keyword feature, and therefore produce inadequate feature representations for a specific domain topic or a topic with high complexity. Our method extracts three types of features, including keyword feature, subject word feature, and domain named entity recognition feature. This makes our topic clustering results more accurate.

B. TOPIC DETECTION AND TRACKING FOR CYBER SECURITY

Cyber security researchers use data mining and machine learning techniques to study threat intelligence [19]. Besides Twitter and micro-blog, research on topic detection has focused on data from news, forums and other standardized platforms [20].

Deliu *et al.* identified emerging threat intelligence in real time using data from hacker forum and blogs [4]. First, they used a support vector machine to classify data sources and filter data that was unrelated to the security domain; then the topic model clustered the security datasets to obtain various kinds of threat intelligence events. Deliu's method directly uses the topic model to cluster security domain data to obtain threat events, which does not consider relevant characteristics of security domain data.

Li *et al.* proposed the OSIF framework, which automatically analyzed open source data to generate a chain of threat intelligence events [5]. First, security event related articles were extracted from the open source platform according to the given event keywords and preprocessed. Then the article entities were extracted, and abstracts were generated. The articles were then sorted chronologically to form a threat intelligence event chain. This method directly extracts the article related to the security event by using a predetermined event keyword. The method is simple, but the extracted articles depend on the given keyword. Some articles related to the security event are easily omitted or mistakenly extract articles related to other events.

Based on general topic detection technology, this paper deeply studies feature extraction methods applied to the security field, improves upon existing topic clustering models, and proposes a threat topic discovery method tailored to the security field.

III. DTD-FETC SYSTEM ARCHITECTURE

The proposed DTD-FETC topic discovery method is mainly composed of the data preprocessing, the features extraction, and the topic clustering. Figure 1 shows the system architecture of DTD-FETC. Based on DTD-FETC method, we build a cyber threat topic discovery system to identify emerging cyber threat topics from open source security news platforms and blogs.

First, users define several domain-related keywords as label categories. The system periodically crawls webpages with the defined label categories from security-related websites, and then extracts the article title, body text, time, and tag category information into a database. Next, the system performs preprocessing operations, including deduplication, deletion of stop words, punctuation, part-of-speech tagging, removal of specific part-of-speech words, and case conversion on the article content and title content. After the preprocessing, the system gets the candidate keywords for the article. Then, the system uses keyword feature extraction method, subject word feature extraction method, and named entity recognition method to extract different types of features

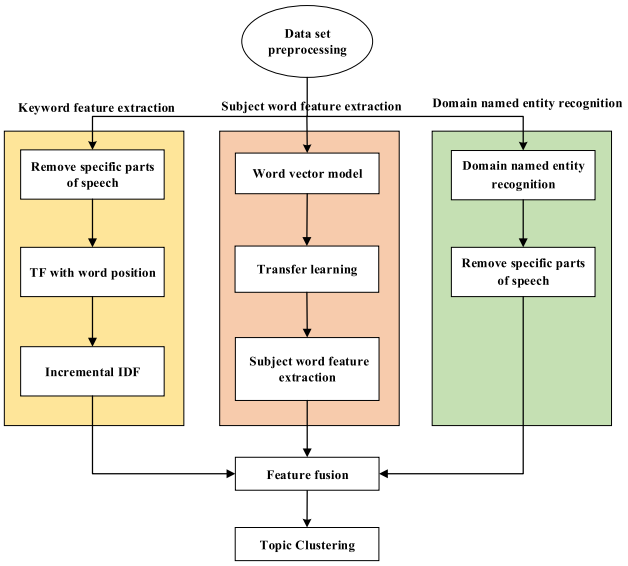

FIGURE 1. DTD-FETC system architecture.

respectively. The three types of features are further fused to construct the feature vector of the article. Finally, the system uses an improved hierarchical clustering algorithm to cluster the feature vectors of articles in each period to identify emerging or historical topics in real time.

In our study, the data is taken from different security news web sites and security information platforms. The data crawling module crawls all the articles of specific label categories (vulnerability, malware) from these platforms in a multi-process way, and extracts the title, body, time, and label categories in the articles, then stores them in the database.

The preprocessing module performs pre-processing operations such as deduplication, stop word deletion, punctuation removal, part-of-speech tagging and removal, lemmatization and case conversion on the body and the title content of article crawled in real time, and then generates candidate words for the feature extraction module.

In this paper, we explore existing feature extraction methods in the security domain. Three improved feature extraction methods are then proposed: an improved keyword feature extraction method, a subject word feature extraction method and an entity feature extraction method. Feature fusion technology is used to merge the resulting features, and the feature vector of the article is constructed for use as the input of the topic clustering module.

To address the problematic ''clustering effect'' that occurs when the hierarchical clustering algorithm detects threat topics, we propose an improved hierarchical clustering algorithm based on the similarity of vector products. Then, the improved clustering algorithm is used to cluster the articles in each period and identify emerging topics in real time.

IV. FEATURES EXTRACTION

A. KEYWORD FEATURE EXTRACTION

Simple and efficient, the TF-IDF (term frequency–inverse document frequency) algorithm is often used for keyword extraction. However, the TF-IDF algorithm has some significant shortcomings. For example, the inverse document datasets in TF-IDF is constant. Since the open source web datasets are dynamically changed, the fixed IDF set does not represent the inverse document frequency of the words in the dynamically changing dataset.

To mitigate these problems, we propose an improved keyword feature extraction method, the ITFIDF-LP (Incremental TF-IDF method considering word location and part of speech) method. The details of this method are described below:

1) THE PART OF SPEECH

The ITFIDF-LP method proposed in this paper first removes the stop words from the article. Similarly, since parts of speech such as qualifiers and quantifiers cannot be keywords, parts of speech such as adverb, numerals, coordinating conjunction, determiner, preposition, comparative adjectives, modal auxiliary, personal pronoun, predeterminer in the article are removed. The remaining words are candidates for keyword feature extraction.

2) TF METHOD CONSIDERING WORD POSITION

Term Frequency (TF) indicates how often the term t appears in document d. The traditional TF method does not consider the influence of a word's position in a document and part of speech of the word, which leads to many words representing the topic information of the article being mistaken for non-keywords by the algorithm [21].

Our proposed method must account for the differing importance of words in different respective positions, such as the distinction between words in the title and body of the article. If a term is in the title, the term has a higher TF value. To that end, we propose a TF method based on the position of candidate words. The formula to calculate the new TF value is as follows (1):

$$tf\ (t,d) = \begin{cases} w_t * TF\ (t,d) & if\ t \in T \\ TF\ (t,d) & if\ t \in C \end{cases} \tag{1}$$

where $TF\ (t,d)$ is the frequency at which the word t appears in document d, T is a set consisting of the title words, C is a set consisting of words in the article body. $TF\ (t,d)$ is calculated by the classic TF-IDF mdethod. w_t is the weight of word in title. The larger w_t indicates that the importance of words in the title is larger. Yu improved the classic Textrank algorithm based on the article structure, such as title and paragraph [22]. According to Yu's research results and the experimental of the keyword extraction method, the value of w_t is set as 1.2.

3) INCREMENTAL IDF METHOD

IDF represents the inverse document frequency of word t. If there are fewer documents containing the term t, the inverse document frequency of the term t is larger, which indicates that the term t has a good class discrimination ability.

TABLE 1. The meaning of symbol in the formulas.

Symbol	Meaning
t	a term
T	a set consisting of the title words
C	a set consisting of words in the article body
d	a document
k	total of documents
w_t	weight of the title words
N_c	total number of documents during the current time period
$n(t,c)$	number of documents containing the word t during the current time period
θ	topic distribution for articles
φ	word distribution
z	topics
w	feature word vector to a document
α	parameter of θ
β	parameter of φ
$S(t)$	similarity of the candidate subject word t to the label category of the article
$L(t)$	weight of the location of t
$P(t)$	weight of the part of speech of t
$weight_{LDA}(t)$	weight of t calculated by LDA model
$weight_{LDA-SLP}(t)$	weight of candidate subject words by LDA-SLP model
c_s	topic vector
a_i	feature vector of the article i within a topic

An incremental IDF method can be used to solve problems caused by an IDF that does not change dynamically with the dataset. The formula to calculate the incremental IDF is as follows (2):

$$idf\ (t,c) = log(\frac{N_c}{n\,(t,c)+1}) \tag{2}$$

where N_c indicates the total number of documents in the database during the current time period, $n\,(t,c)$ is the number of documents containing the word t during the current time period. Since the datasets in the database are dynamically changed, N_c and $n\,(t,c)$ are dynamically changed over time. Based on the improved TF-IDF method described above, the weighting formula for the keyword feature candidate words in the article are calculated as follows (3):

$$weight\ (t,d) = \frac{tf\ (t,d) * idf\,(t,c)}{\sqrt{\sum_{t'\in d}(tf\ (t',d) * idf\,(t',c))^2}} \tag{3}$$

In this paper, the weights of the article's keyword feature candidate words are calculated according to the above formula. The key candidate words of the article are sorted according to the weight, and the first X words are selected as the keywords feature of the article. Table 1 explains the symbolic representations involved in the formulas.

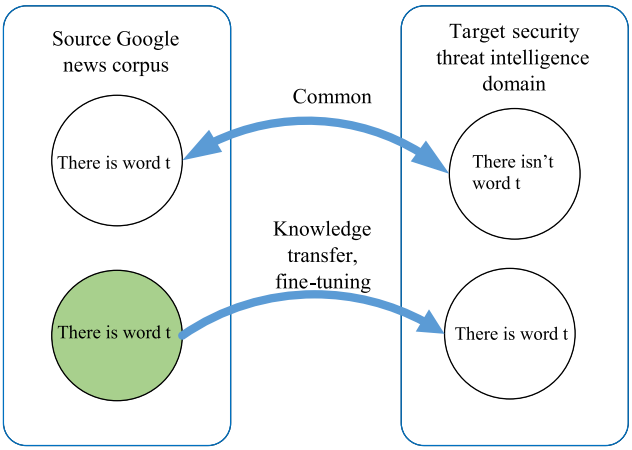

FIGURE 2. Word vector model combined with transfer learning.

B. SUBJECT WORD FEATURE EXTRACTION

1) WORD VECTOR MODEL COMBINED WITH TRANSFER LEARNING

The word vector trained by the Word2Vec model contains the word's semantic information and can reflect the linear relationship between words [23]. Therefore, this paper trains the word vector of the article words based on the Word2Vec model, so as to calculate the similarity in word vectors.

Since security related datasets are sparse, a Word2vec model trained with such a dataset cannot obtain the feature word vector with high-quality semantic information. Therefore, this paper proposes combining this word vector training method with the transfer learning method. First, the word vector pre-training is performed using the Word2vec model on the wiki news corpus. The wiki news corpus is similar to news in the security field, so that the trained Word2vec model has a priori knowledge, and therefore the word vector has a priori parameters. We then transfer the model and parameters to the security threat intelligence dataset for retraining, so that the feature word vector obtained incorporates more relevant semantic information.

As shown below in figure 2, the process of training the feature word vector in Word2vec model is combined with transfer learning. The word vector parameters of words that exist in the wiki news corpus but do not exist in the security threat intelligence data are directly transferred to the model of the target domain. The word vectors of words existing in both the wiki news corpus and threat intelligence domain are transferred to the model of the target domain, and then security threat intelligence data is added to fine-tune the word vector parameters of these words.

2) SUBJECT WORD FEATURE EXTRACTION METHOD BASED ON TOPIC MODEL

LDA is the most commonly used topic detection model, which mainly uses the article subject probability distribution and the subject word probability distribution to obtain final candidate subject words for the document [24].

TABLE 2. Label category and candidate subject word vector similarity.

label category	candidate subject word	label category and candidate subject word vector similarity
vulnerability	affect	0.0771
vulnerability	eternalblue	0.3135
malware	researcher	0.1194
malware	trojan	0.7067

The shortcoming of the LDA model for subject word extraction is that high-frequency but irrelevant words are easily selected as candidate subject words. We need to remove these words from among the candidate subject words in order to obtain final subject feature words which are closely related to the subject.

The vulnerability and malware category articles on the target security websites contain specific threat information. The research goal of this paper is to identify the threat topic. Therefore, this paper uses vulnerability and malware category news from the security website as its data source. The label category of each article in the data source is vulnerability or malware.

In general, the subject of the article is closely related to the label category of the article. The greater the similarity of the candidate subject word to the label category of the article, the stronger the indication that the candidate subject word is more closely related to the subject of the article. Therefore, this paper uses a word vector model based on transfer learning to obtain the feature word vector. The similarity between the candidate subject word vector and the label category word vector is then calculated. The similarity is used as the coefficient of the candidate subject words' weight, then the candidate subject words weights are recalculated, and some subject words are filtered. Table 2 shows examples of similarities between candidate subject terms and label category word vectors in the article.

The article candidate subject words obtained by using the LDA topic model contain many high-frequency words that are irrelevant to the subject, such as the candidate subject words 'affect' and 'researcher' in Table 1. Irrelevant words can impede the effective feature extraction of article subject words. Therefore, the weights of these words are reduced by calculating the similarity between the label category and the article candidate subject vector. For example, the similarity between the word 'affect' and the label category in the above table is 0.0771. The similarity is taken as the coefficient of the candidate subject word weight. Then the weights of such words are reduced, and such words are filtered in the process of extracting subject feature words.

The part of speech and location of the candidate subject words can determine the importance of the candidate subject words, thereby affecting the extraction of the final subject feature words. Since candidate subject words that are noun and proper noun parts of speech are more likely to be the topic word of the article, our method increases the weight of candidate subject words that are those parts of speech. The title of the article contains more significant subject information than the body of the article, so our method increases the weight of the candidate subject words included in the title.

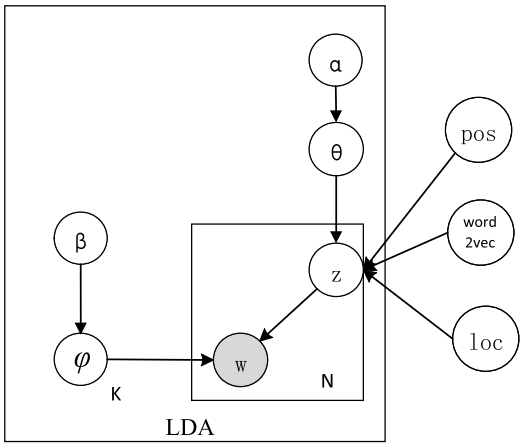

FIGURE 3. Subject words feature extraction model.

Figure 3 shows the subject word feature extraction model LDA-SLP (LDA-word vector Similarity & Location & Part of speech) we proposed. As shown in figure 3, the LDA model in the large rectangle extracts the article candidate subject words and gets its weight. The similarity between candidate subject words and article category label, part of speech, and word position of the candidate subject words on the right side jointly determine the weight of the candidate subject words.

The purpose of the LDA model is to generate the document topic distribution θ and topic word distribution φ for each article. z is the subjects and is generated by θ. W is feature word vector to a document. α is parameter of θ and β is parameter of φ. Griffiths and Steyvers proposed Gibbs Sampling (GS) method for approximate estimation of α and β of LDA model [25]. We use GS method to get α and β.

The weights of candidate subject words in LDA-SLP are calculated as follows (4):

$$\begin{aligned} &weight_{LDA-SLP}(t) \\ &\quad = (K_1 \cdot S(t) + K_2 \cdot L(t) + K_3 \cdot P(t)) \times weight_{LDA}(t) \\ &L(t) = \begin{cases} 1, & t \text{ is in content} \\ 2, & t \text{ is in title} \end{cases} \\ &P(t) = \begin{cases} 0, & t \text{ is other parts of speech} \\ 1, & t \text{ is noun} \end{cases} \end{aligned} \tag{4}$$

where $S(t)$ is the similarity of the candidate subject word t to the label category of the article, $L(t)$ is the location of t, $P(t)$ is the part of speech of t, $weight_{LDA}(t)$ is the weight of t calculated by LDA model. The values in $L(t)$ is set based on Yan's study [26]. Because position, similarity, and part of speech of a term play different roles in determining the weight of a candidate subject word, they are assigned different weights, K_1, K_2 and K_3. Generally, we can set

$K_1 = K_2 = K_3 = 1$, or adjust the values of K_1, K_2 and K_3 according to the experimental results.

The candidate subject words of each article are sorted in reverse order according to the weight, and the first N words are selected as the article subjects z.

C. DOMAIN NAMED ENTITY RECOGNITION

A topic is a collection of articles that tell one or more related events in chronological order around a person, place, or organization [1]. Therefore, the person, places and organizations entities of the article are important features in the topic detection model. In recent years, many statistical models and deep learning models have been applied to entity recognition [27]. Taking conditional random fields as an example, the results of entity recognition are mainly determined by their feature functions and parameters. The feature functions are mainly defined by the part of speech and the relationship of the words in the context. Due to the innate variety of words, entities, and parts of speech, general entity recognition accuracy is not high.

To address those problems, this paper proposes the following two solutions with particular consideration of the relevant characteristics of the network security field.

1. This article restricts the extracted entities to only person, place, and organization. These entity words appear in the dataset with high frequency, contain few of the excluded parts of speech, and have an obvious context structure, so these entities have higher recognition accuracy.

2. Since determiner, adverbs, numerals and coordinating conjunction part-of-speech words by nature cannot be the entity feature words we are concerned with, they are removed from the recognized entity words, and the remaining words identified as the person, place and organizational entity are used as extracted entity features to improve the entity recognition accuracy.

D. FEATURE FUSION

Feature fusion is performed to obtain the feature vector of the article. Each dimension of the feature vector represents the weight value of a feature word, that is, the importance, and the feature vector of the article is used as the input of the topic cluster. The weight value of each feature word in the article feature vector is determined by the above-mentioned keywords, subject words, and entity feature extraction methods. First, the feature words extracted by the three feature extraction methods are normalized. Then, the normalized feature words are weighted, and the vector composed of all feature word weights is used as the final article feature vector.

V. TOPIC CLUSTERING MODEL BASED ON VECTOR PRODUCT SIMILARITY

A. HIERARCHICAL CLUSTERING METHOD

The hierarchical clustering method has the characteristics of not needing to set the number of topics in advance [28]. Instead, it offers a more dynamic approach. The hierarchical relationship of classes can be found, and the convergence conditions can be easily set, making it a superior

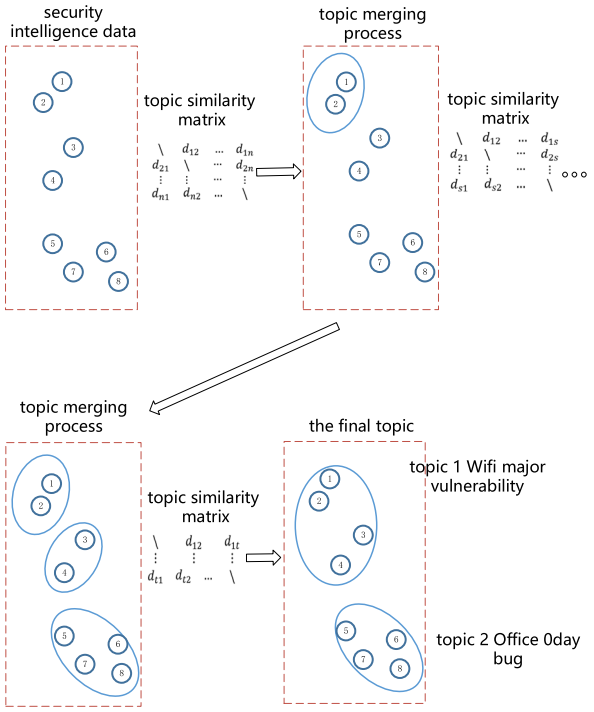

FIGURE 4. Hierarchical clustering process.

method for a variety of clustering contexts. Due to these advantages, we use an hierarchical agglomerative clustering method in this paper. Based on hierarchical clustering, hierarchical agglomerative clustering is an improved topic clustering model that better detects emerging threat topics in real time.

Figure 4 shows the process of clustering topics on some security related web pages by an agglomeration hierarchical clustering algorithm.

Step 1) The convergence conditions are set, with each article within the time period Δt set as a security topic cluster.

Step 2) The similarity matrix between topics is constructed, and the two most similar topics are selected to merge into a new topic Z.

Step 3) The similarity between other topics and the new topic Z is recalculated and updated, then repeatedly merged until the similarity of any two topics is lower than the set threshold, leaving only the final few topics.

The similarity between the two most similar topics becomes smaller and smaller as the clustering progresses. The clustering threshold is set as the percentage of the maximum similarity between topics, at a value of 0.25. The absolute threshold needs to be dynamically adjusted as the dataset changes. The optimal value of the relative threshold is 0.25 for several datasets used in this paper. It shows that the convergence condition is set as the relative threshold for topic clustering and has good robustness and adaptability.

B. CLUSTERING EFFECT

As mentioned above, the key step in the hierarchical clustering is to construct a similarity matrix between topics.

The topic similarity calculation methods include single linkage, average linkage, weighted linkage, and centroid linkage methods. The single, average or weighted linkage methods require calculating the similarity of each data point between topics, and the computational complexity is $O(n^2)$. The centroid linkage method takes the topic centroid vector as the topic vector, and uses the topic vector similarity to measure the similarity between two topics, and the computational complexity is $O(n)$. This makes the centroid linkage method better suited for real-time threat topic discovery. To better maintain real-time feasibility, we used the centroid linkage method to calculate the similarity between topics through hierarchical clustering.

The traditional centroid linkage method is governed by the following concepts:

1) The topic vector is the arithmetic mean of the corresponding dimension elements to all article feature vectors within the topic.

$$c_s = (a_1 + a_2 + a_3 \ldots + a_k)/k \tag{5}$$

where a_i is the feature vector of the article i within the topic, and k is the total of articles.

2) The topic vector similarity is used to represent the topic similarity, and the topic vector similarity is measured by the angle of the topic vector.

The centroid linkage method uses topic vector similarity to measure topic similarity. Topic vectors similarity is often measured using the cosine of the angle between the topic vectors. Using the cosine of the angle between the vectors to measure the similarity of the vectors, it easily leads to the ''clustering effect'', that is, the similarity of the data in the cluster is low.

The feature vector dimension of the article is 1-N dimension. Figure 5 shows the topic merging process when the dimension of the article feature vector is assumed to be two dimensions. The black point in the figure is the feature vector of each article. Each of the ellipses is a topic, the gray point V_1 is the topic vector of cluster C_1, the gray point V_2 is the topic vector of cluster C_2, and the gray point V_3 is the topic vector of cluster C_3 in the figure. Because $\theta_1 < \theta_2$, the traditional centroid linkage method combines the C_1 and C_2 into a new cluster C_4. The rectangle in the figure is C_4, and V_4 is the new topic vector of C_4. It can be seen from the figure that some articles in C_4 are far away from each other and have low similarity. This is the phenomenon of ''clustering effect''.

C. VP-LINKAGE: CENTROID LINKAGE METHOD BASED ON VECTOR PRODUCT

''Clustering effect'' phenomenon is due to the fact that the traditional centroid linkage method merges topics by considering only the angle between topic vectors. However, the angle between the topic vectors does not guarantee that the articles between the new topics are similar.

Since the length of the feature vector does not have a fixed range, the feature vector module of each article is initially

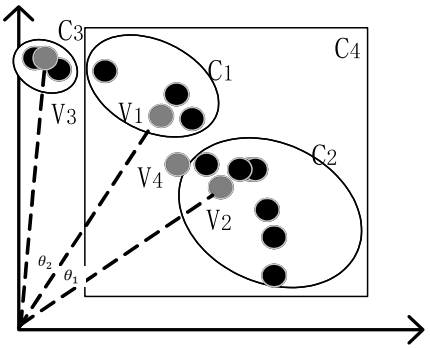

FIGURE 5. Centroid linkage method.

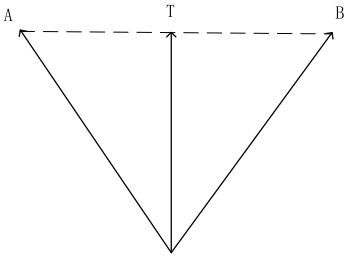

FIGURE 6. Article vector merging.

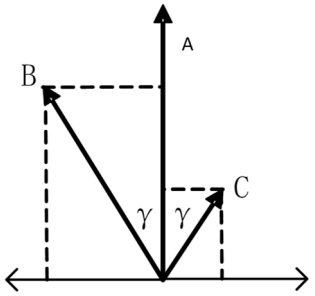

FIGURE 7. Vector similarity comparison.

normalized to 1. The topic vector is the arithmetic mean of the corresponding dimension elements to all article feature vectors within the topic, so topic vectors are generally shorter than article vectors. In figure 6, the topic centroid vector of the article vectors $\vec{A}$ and $\vec{B}$ is $\vec{T}$, and the length of $\vec{T}$ is smaller than that of $\vec{A}$ and $\vec{B}$. The length of topic centroid vector $\vec{T}$ is related to the similarity of $\vec{A}$ and $\vec{B}$. If the two article vectors are similar, the length of the topic vectors of the two articles is also longer. The smaller the similarity of the two article vectors, the shorter the length of the topic vector. Therefore, we can get Lemma 1.

Lemma 1: The module of the topic vector can relatively represent the similarity between the articles in the topic. The longer the topic vector, the higher the similarity of the articles in the topic.

With the aggregation of article vectors, the lengths of the generated topic vectors become unequal. As shown below in figure 7, the angle between topic vectors $\vec{A}$, $\vec{B}$ and between

$\vec{A}$, $\vec{C}$ is equal. The traditional centroid linkage method measures the similarity between topics A and B and determines if it is equal with the similarity between topics A and C. That is, the probabilities that topic A and topic B or topic A and C are merged into a new topic are equal. However, the Euclidean distance between topic vector $\vec{A}$ and topic vector $\vec{B}$ is smaller because the article feature vectors in the topics are distributed around the topic vector. This means that the Euclidean distance between the article feature vectors in topic B and the article feature vectors in topic A is small. A reasonable topic merging process should dictate that topic A and topic B be merged into a new topic.

In this way, both the length of a topic vector and the angle between two vectors are related to the similarity of the vectors. Therefore, in addition to considering the angle between the topic vectors, this paper also sees fit to consider the module of the topic vectors. The vector product formula includes the angle between the two vectors and the length of the two vectors, so vp-linkage method was proposed to calculate the topic similarity based on the vector products. The equation (6) is the calculation formula for topic similarity:

$$Similarity\,(D_A, D_B) = \vec{A} \cdot \vec{B} = \left|\vec{A}\right| \cdot \left|\vec{B}\right| \cdot \cos\theta \qquad (6)$$

where D_A is the topic A, $D_B D_B$ is the topic B, $\vec{A}$ is the topic vector of A, $\vec{B}$ is the topic vector of B, and θ is the angle between the two topic vectors.

When the vector product is used to represent the similarity of topic vectors, in the case where the included angles between the topic vectors are approximately the same, the algorithm will preferentially select topic vectors with a longer length for merging. In this way, topic vectors with a small number of merged times and high similarity of articles within the topic are selected to be merged, so that the phenomenon of too many articles in a topic can be avoided. The topic clustering algorithm based on vp-linkage can maintain the merging times of most topics in a relatively average state, and make the articles in the topic have higher similarity. These strengths allow it to obtain a more stable clustering effect. This method both retains the advantage of the low complexity of the centroid linkage method to measure the similarity between two topics and also avoids the centroid linkage method's shortcomings regarding its negative tendency to form a "clustering effect."

VI. DATASETS AND EVALUATION METHODS

A. DATASETS

1) DATASET 1

The open source English wiki news corpus is about 13 GB (*https://dumps.wikimedia.org/enwiki/latest/*). It comprises more than 100 million sentences, which are pre-trained datasets for the word2vec model.

TABLE 3. 12 malware threat topics.

Topic	Count
Australia, Canada, Others Blame North Korea for WannaCry Attack	17
New Windows Trojan Spreads MIRAI Malware To Hack More IoT Devices	14
Carberp loading: New generation of financial malware on the rise	11
Triton Malware Exploited Zero-Day in Schneider Electric Devices	7
Dyre Wolf Banking Malware Stole More Than $1 Million	8
VPNFilter Continues Targeting Routers in Ukraine	7
Duqu Trojan developed in unknown programming language	23
Destructive Rombertik Sample Traced Back to Nigerian Man: ThreatConnect	5
Shamoon 2 Used Rudimentary Method for Network Distribution	11
XML Files Used to Distribute Dridex Banking Trojan	15
XcodeGhost Compiler Malware Targets iOS, OS X Systems	7
CryptoWall 2.0 Ransomware Capable of Executing 64-Bit Code: Cisco	14

B. DATASET 2

We used a crawler tool to collect all data from 8 security BBS, blogs, and news platforms, such as *www.blackhat.com, www.darkreading.com, www.securityweek.com, www.technewsworld.com, www.beyondtrust.com/blog*. Dataset 2 contains approximately 30,000 articles and 800,000 sentences. These sentences were used as datasets for retraining the word2vec model.

C. DATASET 3

We extracted 139 articles from the collected malware label articles for annotation. The annotation process was cross-labeled by 3 people, and resulted in 12 threat topics as Table 3 shows. The annotation malware label data was used to evaluate the results of the security threat intelligence topics discovery.

D. DATASET 4

We extracted 150 articles from the collected vulnerability label articles for annotation, resulting in 8 security threat topics as Table 4 shows. The annotation vulnerability label data was used to evaluate the results of the security threat intelligence topic discovery.

TABLE 4. 8 vulnerability threat topics.

Topic	Count
Fileless Ransomware Spreads via EternalBlue Exploit	9
Microsoft Takes Steps to Protect IE Users Against POODLE Attacks	8
New Variants Found in Spectre and Meltdown	15
Oracle Releases Patches for Exploited Apache Struts Flaw	18
OpenSSL Patches Flaws Found With Google Fuzzer	40
Android Stagefright Exploit Released	9
Shellshock Attacks Still Cheap and Easy: IBM	8
Adobe Flash Zero-Day Under Attack	23

TABLE 5. The keywords by extracted TF-IDF and ITFIDF-LP.

Article	TF-IDF	ITFIDF-LP
Australia,Canada, Others Blame North Korea for WannaCry Attack	North, Korea, WannaCry, said, United, other	korea, north, wannacry, canada, consultation, united
VPNFilter: New Exploit Feature and Affected Devices Revealed	routers, into, devices, endpoints, thought, than	router, endpoint, affected, reveal, vpnfilter, device
Source Code Released for Mirai DDoS Malware	said, devices, attacks, DDoS, there, they	ddos, botnets, arbor, dobbin, krebs, mirai
Inside Cryptowall 2.0 Ransomware	Cryptowall, was, said, Carter, Tor, could	carter, cryptowall, 64bit, virtual, tor, amd

E. EVALUATION METHODS

We evaluated, recall rate, accuracy and F value, which are widely used in text clustering. In this paper, the topic label predicted by the DTD-FETC method is called a cluster, and the manually annotated topic label is called a class. The recall rate, accuracy, and F-value metrics are defined in topic clustering as follows:

$$Recall\ (i,j) = \frac{n_{ij}}{n_i}$$
$$Precision\ (i,j) = \frac{n_{ij}}{n_j} \tag{7}$$

^{n_i}, ^{n_j}, is the number of articles of class i and cluster j respectively, and n_{ij} is the number of articles containing class i in cluster j. The F value is determined by the recall rate and accuracy. The formula for the F value of cluster j and class i is as follows:

$$F\ (i,j) = \frac{2 * Recall\ (i,j) * Precision(i,j)}{Recall\ (i,j) + Precision(i,j)} \tag{8}$$

The cluster label of clustering indicates which cluster the article belongs to, and it has no actual category meaning. Therefore, the value of F_i of class i in clustering is the largest $F\ (i,j)$ among the $F\ (i,j)$ values of class i and all clusters j. $F\ (i,j)$ is defined as follows:

$$F_i = arg_j max\ (F\ (i,j)) \tag{9}$$

The recall rate R_i and the accuracy P_i of the class i are defined as described above. The final global accuracy, recall, and F value are the weighted average of the accuracy, recall, and F values for each category. The formulas are as follows:

$$Precision = \sum_i \frac{n_i}{n} P_i \tag{10}$$
$$Recall = \sum_i \frac{n_i}{n} R_i \tag{11}$$
$$F = \sum_i \frac{n_i}{n} F_i \tag{12}$$

VII. EXPERIMENT AND RESULTS

The server used in our experiment is as follows: one Intel CORE i9-9900K CPU, one 6T hard disk, 64G memory and one NVIDIA GeForce RTX 1080Ti GPU graphics card.

A. DTD-FETC SYSTEM IMPROVED FEATURE EXTRACTION METHOD AND TOPIC CLUSTERING EXPERIMENT

1) KEYWORD FEATURE EXTRACTION EXPERIMENT

Table 5 shows the keywords of some articles extracted by TF-IDF and our ITFIDF-LP. The keywords extracted by TF-IDF method contain many frequent words. For example, the keywords extracted by TF-IDF method include 'said', 'other', 'than', 'was' and 'they'. These common words do not represent the topic of the article. The keywords extracted by ITFIDF-LP keyword extraction method do not include these common words. The keywords such as 'mirai', 'botnet' and 'tor' are more relevant to the topic of the article.

As shown in Table 6, selecting the number of different keyword features will influence the effect of keyword feature extraction. For dataset 3, the number of keyword features K was set to 5, and the number of key features selected in each document is too small. As a result, articles originally belonging to the same topic are separated when clustering threat topics. With K set to 20, too many keyword features were selected in each article, and the threat topic detection often exhibited clustering errors; when K was set to 10, the threat topic clustering result based on keyword features performed the best. For dataset 4, K was 20, and the threat topic clustering based on keyword features achieved good results.

As shown in Table 7 below, this study compared keyword feature extraction methods that both include and exclude the title of the article. The experimental results from datasets 3 and 4 indicated that keyword feature extraction methods which include the article title yield better threat topic clusters.

TABLE 6. The influence of keyword quantity parameter on experiment.

Datasets	keyword quantity	F1	Precision	Recall
dataset3	5	0.876	0.980	0.815
	10	**0.961**	**0.980**	**0.935**
	20	0.957	0.990	0.920
dataset4	5	0.840	0.977	0.748
	10	0.923	0.983	0.880
	20	**0.937**	**0.990**	**0.931**

TABLE 7. The impact of article title on experiment.

datasets	article title	F1	Precision	Recall
dataset3	**contain the title**	**0.961**	**0.980**	**0.935**
	without the title	0.930	0.980	0.902
dataset4	**contain the title**	**0.937**	**0.990**	**0.931**
	without the title	0.839	0.981	0.819

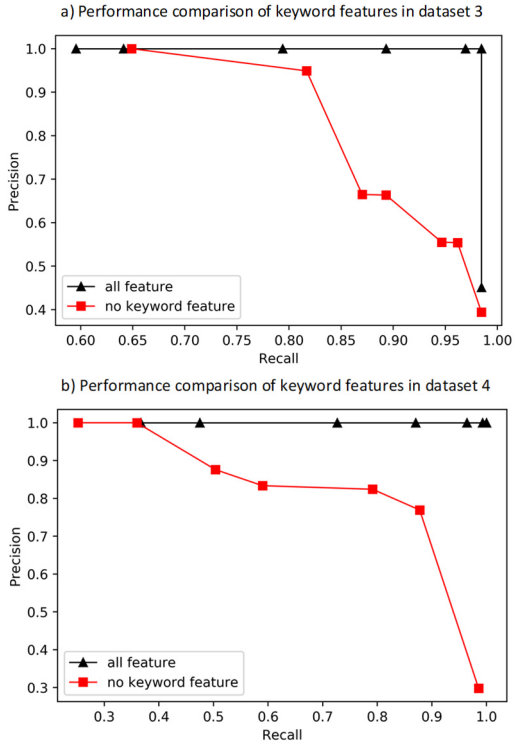

FIGURE 8. Keyword feature performance comparison p-r curve.

As shown in figure 8, based on dataset 3 and dataset 4, this study compared impact of the presence and absence of keyword features on threat topic discovery system performance. The result of precision-recall rate curve shows that keyword feature played an important role in improving threat topic detection effectiveness.

2) SUBJECT WORD FEATURE EXTRACTION EXPERIMENT

Table 8 shows the subject words of some articles extracted by LDA and LDA-SLP. The subject words extracted by the LDA model contain a lot of high-frequency words, which

TABLE 8. The subject words extracted by LDA and LDA-SLP.

Article	LDA	LDA-SLP
WannaCry Does Not Fit North Korea's Style	say, malware, attack,wannacry, security, code	wannacry, attack, ransomware, malware, code, security
Three Hackers Plead Guilty to Creating IoT-based Mirai DDoS Botnet	malware, use, say, researcher, attack, system	dridex, new, campaign, attack, malware, system
Source Code Released for Mirai DDoS Malware	malware, attack, use, say, target, device	mirai, attack, ddos, malware, botnet, target
Inside Cryptowall 2.0 Ransomware	malware, say, file, use, security, researcher	malicious, cryptowall, file, malware, ransomware, security

TABLE 9. The influence of word vector similarity parameter on experiment.

datasets	word vector similarity parameters	F1	Precision	Recall
dataset 3	**after introduction**	**0.660**	**0.767**	**0.690**
	before introduction	0.359	0.81	0.417
dataset 4	**after introduction**	**0.753**	**0.922**	**0.740**
	before introduction	0.589	0.701	0.625

do not express the meaning of the article well. For example, the subject words extracted by the LDA topic model include 'say', 'use' and 'researcher'. The candidate subject words extracted by LDA-SLP model are determined by the similarity between candidate subject words and article category label, part of speech, and position of the candidate subject words. So, the high-frequency candidate subject words are removed. The subject words extracted by LDA-SLP include 'cryptowall', 'ransomware', 'mirai' and 'ddos', which reflect the subject of the articles.

As shown in the Table 9, we applied the label category and feature word vector similarity calculation method to improve the subject word feature extraction method. The evaluation of the threat topic discovery results was based on the subject word feature extraction method before and after the improvement. The experiments show that the improved method yielded an improvement in results from the threat topic discovery system. Because a single subject has a single feature, the threat topic discovery using only the subject word feature is not effective.

Selecting the number of different subject words in the subject word feature extraction method affected the threat topic discovery results. Table 10 shows that, for datasets 3 and 4,

TABLE 10. The influence of the number of subject words on experiment.

datasets	subject word quantity parameter	F1	Precision	Recall
dataset 3	6	0.690	0.767	0.660
	10	**0.741**	**0.769**	**0.640**
	20	0.611	0.817	0.573
dataset 4	**6**	**0.753**	**0.922**	**0.740**
	10	0.730	0.790	0.748
	20	0.719	0.695	0.877

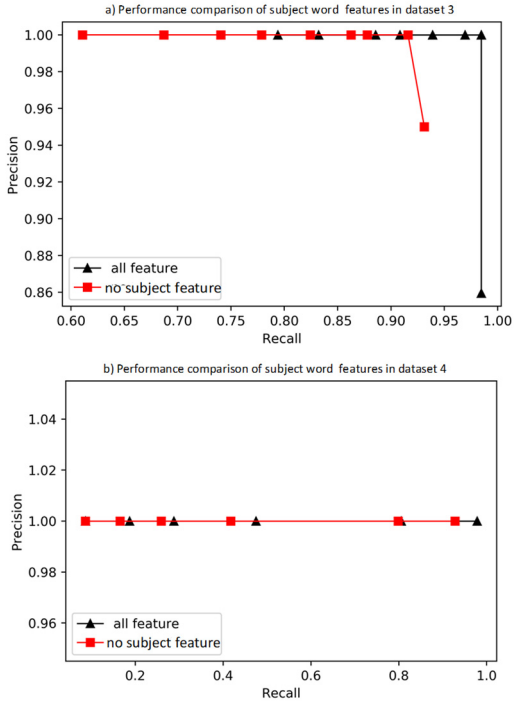

FIGURE 9. Subject word feature performance comparison p-r curve.

the range of subject words K between 6 to 10 achieved good results. With K at 20, too many high-frequency words irrelevant to the subject feature of articles were extracted, so threat topic detection clustering errors occurred. In the above experiments, the WPGMA (Weighted Pair Group Method with Arithmetic Mean) hierarchical clustering method was used to cluster threat intelligence topics [29].

As shown in figure 9, based on dataset 3 and dataset 4, we compared the influence of the presence or absence of subject word features on threat topic discovery system performance. The precision-recall rate curve indicates that dataset 4 used keyword and entity features to perform threat topic detection and reach a precision and recall rate of 100%. Increasing subject word features in dataset 4 greatly improved the results. The subject word features improved the threat topic discovery results from dataset 3 as well.

TABLE 11. Some extracted named entities.

Article	Named entities
Kaspersky perplexed by Duqu code	duqu, framework, kaspersky, microsoft, payload
Carberp loading: New generation of financial malware on the rise	amit, carberp, facebook, klein, trojan, trusteer
Dridex Returns With Windows UAC Bypass Method	australia, campaign, dridex, mcafee, microsoft, ddos
XML Files Used to Distribute Dridex Banking Trojan	dridex, flashpoint, kremez, trojan, u.k, uac, window

3) ENTITY FEATURE EXTRACTION EXPERIMENT

Each topic or event generally has a specific character, organization, and location. Therefore, the named entity we extracted are characters, places, and organizations in an article. Table 11 shows some named entities. Named entities, such as companies in a topic, enhance the domain characteristics of the topic. For example, 'mcafee', 'kaspersky' and 'microsoft' are well-known companies in cyber security.

As shown in figure 10, a) and b) are the experimental results of the threat topic discovery system with and without entity features from both dataset 3 and dataset 4. The precision-recall rate curve indicates that the entity feature improved threat topic detection effectiveness. Entity features were complementary to keywords and topic word features, with the three features combining to form the topic feature vector of the article, improving detection results.

4) EXPERIMENT BASED ON HAC IMPROVED TOPIC CLUSTERING MODEL

As shown below in Table 12, the topic clustering methods before and after the improvements were experimentally verified using dataset 3 and dataset 4. The relative threshold is 0.25 in the experiment. Our experiments show that the hierarchical clustering algorithm based on a centroid linkage method is easily formed a ''clustering effect'', yielding poor clustering results. The weighted pair-group with arithmetic means (WPGMA) hierarchical clustering algorithm can yield better results, but the results from the proposed improved HAC method indicate that it can yield the best results.

Figure 11 shows a) and b), the experimental results of the threat topic detection before and after implementing the improved HAC clustering method on dataset 3 and dataset 4 respectively. The precision-recall rate curve indicates that the improved hierarchical clustering method greatly improved threat topic discovery system discovery and obtained more stable clustering.

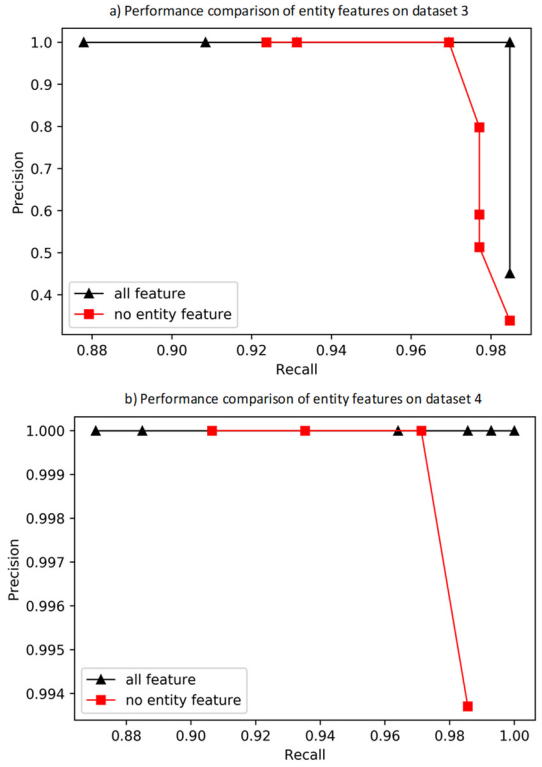

FIGURE 10. Entity feature performance comparison p-r curve.

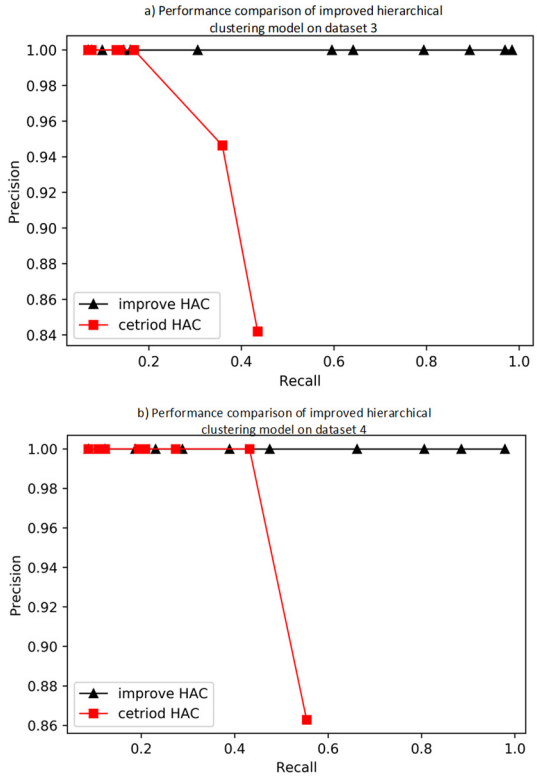

FIGURE 11. Improved HAC method performance comparison p-r curve.

TABLE 12. HAC Performance comparison.

datasets	Clustering algorithm	F1	Precision	Recall
dataset 3	HAC of centroid linkage method	0.543	0.902	0.501
	vp-linkage HAC	**0.999**	**0.999**	**0.999**
	WPGMA HAC	0.965	0.961	0.941
dataset 4	HAC of centroid linkage method	0.290	0.841	0.435
	vp-linkage HAC	**0.992**	**0.999**	**0.984**
	WPGMA HAC	0.946	0.985	0.910

TABLE 13. DTD-FETC parameters.

Type	Parameter	Value
Keyword feature	Number of keywords	10
	Title: Body content weight ratio	1：1
Subject word feature	Number of subject words	10
	K_1，K_2, K_3	2, 3, 1.5
Entity feature	Entity	Person, place, organizational entity
Feature fusion	Keyword feature：Subject word feature：Entity feature	0.5:0.2:0.5
Improved HAC algorithm	clustering relative threshold	0.25

B. COMPARISON OF THREAT TOPIC DISCOVERY METHODS

The experiments in section 7-A proved that the number of keywords, the number of topic words in the article, the part-of-speech of the candidate topic words, the similarity of feature words and tag categories, and whether the candidate topic words are heading words will affect the experimental results. The experiment in section 7-A-1 proves that when the number of keywords is about 10, and the ratio of the number of keywords in the title and the text is 1: 1, the experimental results are the best. The number of topic words for each topic is optimized through experiments. Experiments show that the effect is best when the number of topic words is about 10.

In the process of system joint adjustment of parameters, the main task is to adjust the weight of each feature in the feature fusion stage. The detailed parameters of the various algorithms in the DTD-FETC system are shown in the following Table13.

We compare our approach with the following topic detection methods as baselines. They are (1) AvgDoc, AvgDoc-TFIDF (Jianbo, 2017), (3) HUP cluster (Jiajia Huang, 2015), (4) OAP-HAC (Li Ding, 2018), (5)

LTDMF(Haitao Zheng, 2018), (6) OSIF (Ke Li, 2018) and (7) WEC (Carmela Comito, 2019).

Dataset 3 and dataset 4 are merged together as experimental data. Table 14 shows the experimental results of comparing various topic discovery methods on cyber security dataset. It can be seen from the table that the DTD-FETC method proposed in this paper yielded the best results with the highest precision, recall and F1 when applied to the cyber security datasets.

The word vector trained by the Word2Vec model contained the word's semantic or grammatical information. The average feature word vector AvgDoc and the weighted average feature word vector AvgDoc-TFIDF methods were used to obtain the article feature vector. WEC used Word2Vec model to produce word embeddings as well. Due to the limited security domain content in the training dataset, the trained word vectors by Word2Vec did not have good security semantic information. Therefore, results of AvgDoc and AvgDoc-TFIDF were poor. Since WEC considers both semantic and lexical characteristics of the contents, the result of WEC is better than AvgDoc, AvgDoc-TFIDF.

HUPC topic detection method studies high utility pattern as a feature. OAP-HAC method focuses on improving hierarchical agglomerative clustering algorithms. However, it only adds the position weight of keywords as article feature. The key of OSIF is to extract named entities and use these named entities as cyber threat feature. The above three methods are all single-feature topic detection. These features only represent one aspect of the article, rather than comprehensive features, so the final experimental results are not as good as DTD-FETC.

The features used in LTDMF are temporal feature, geographic feature, co-occurrence feature and hashtag feature, but LTDMF clusters topics based on HAC algorithm. HAC algorithm is prone to the ''clustering effect''. Therefore, the topic clustering result is not good.

Three feature extraction methods extracted different features of the article in DTD-FETC, and an improved hierarchical clustering algorithm was used to cluster the topics. The experimental results were markedly better, suggesting that our proposed DTD-FETC method has rich potential applications to clustering and threat topic discovery systems.

VIII. DISCUSSION

DTD-FETC currently has some flaws. First, DTD-FETC consumes more time than the single-feature extraction method because it needs to extract three different features.

Secondly, DTD-FETC is domain-oriented, but if the domain-related training data is insufficient, the trained word vectors cannot learn good semantic information of the domain.

Third, the named entities we extract are limited to people, organizations and places. These entities cannot comprehensively describe a security event, such as no target system or device being attacked.

TABLE 14. Comparative experiment of different topic detection methods.

methods	F-measure	Precision	Recall
AvgDoc	0.502	0.91	0.454
AvgDoc-TFIDF	0.643	0.912	0.514
HUPC	0.747	0.769	0.726
OAP-HAC	0.829	0.803	0.856
LTDMF	0.77	0.76	0.79
OSIF	0.806	0.849	0.767
WEC	0.935	0.932	0.938
DTD-FETC	**0.996**	**0.998**	**0.992**

Finally, the detected threat topics cannot be directly read and used by security protection equipment. The detected security threat topics are stored in a database and can be read by users. The main function of the system is to find special threat topics in time and provide security staff with more comprehensive introduction about potential threats and attack events. Network security staff does not need to manually browse multiple network security websites to track the latest news and developments in this field.

IX. CONCLUSION

In this study, we proposed a novel DTD-FETC method to analyze open source web data and identify topics in real time. Using this method, we conducted some exploratory work regarding security domain feature extraction methods and topic clustering models. This paper proposed three feature extraction methods, namely the keyword feature extraction method ITFIDF-LP, the subject word feature extraction method LDA-SLP and a named entity feature extraction method. Based on the HAC algorithm of the centroid linkage method, this paper proposed a centroid linkage method based on vector product similarity. Our experiments indicated that the proposed methods yielded greatly improved experimental results. The experimental results showed that the F1, precision and recall of DTD-FETC are 0.996,0.998 and 0.992.

In the future, a security domain entity database can be built to identify various entities in the security domain by using entity identification or classification methods, to improve entity identification results. Also, multi-layer topic clustering structures can be used to identify topics and related events, so as to observe trends more accurately.

REFERENCES

[1] J. Allan, ''Introduction to topic detection and tracking,'' in *Topic detection and Tracking*. Springer, Boston, MA, vol. 2002, pp. 1–16.

[2] Y. Chen and L. Liu, ''Development and research of topic detection and tracking,'' in *Proc. 7th IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Aug. 2016, pp. 170–173.

[3] H. Li and Q. Li, ''Forum topic detection based on hierarchical clustering,'' in *Proc. Int. Conf. Audio, Lang. Image Process. (ICALIP)*, Jul. 2016, pp. 529–533.

[4] I. Deliu, C. Leichter, and K. Franke, "Collecting cyber threat intelligence from hacker forums via a two-stage, hybrid process using support vector machines and latent Dirichlet allocation," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 5008–5013.
[5] K. Li, H. Wen, H. Li, H. Zhu, and L. Sun, "Security OSIF: Toward automatic discovery and analysis of event based cyber threat intelligence," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Oct. 2018, pp. 741–747.
[6] R. McMillan. (May 2013). *Open Threat Intelligence*. [Online]. Available: https://www.gartner.com/doc/2487216/definition-threat-intelligence
[7] W. Tounsi and H. Rais, "A survey on technical threat intelligence in the age of sophisticated cyber attacks," *Comput. Secur.*, vol. 72, pp. 212–233, Jan. 2018.
[8] R. R. Ramnani, K. Shivaram, S. Sengupta, and A. K. M., "Semi-automated information extraction from unstructured threat advisories," in *Proc. 10th Innov. Softw. Eng. Conf. (ISEC)*, 2017, pp. 181–187.
[9] X. Liao, K. Yuan, X. Wang, Z. Li, L. Xing, and R. Beyah, "Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 755–766.
[10] X. Shu, F. Araujo, and D. L. Schales, "Threat intelligence computing," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2018, pp. 1883–1898.
[11] K.-C. Lee, C.-H. Hsieh, L.-J. Wei, C.-H. Mao, J.-H. Dai, and Y.-T. Kuang, "Sec-buzzer: Cyber security emerging topic mining with open threat intelligence retrieval and timeline event annotation," *Soft Comput.*, vol. 21, no. 11, pp. 2883–2896, Jun. 2017.
[12] SITA. (2015). *Threat Intelligence at 30,000 Feet: Securing The Air Transportation Industry*. [Online]. Available: https://go.recordedfuture.com/hs-fs/hub/252628/file-2607572540-pdf/case-studies/sita.pdf.
[13] J. Huang, M. Peng, and H. Wang, "Topic detection from large scale of microblog stream with high utility pattern clustering," in *Proc. ACM Workshop Workshop Inf. Knowl. Manage.*, 2015, pp. 3–10.
[14] C. Comito, A. Forestiero, and C. Pizzuti, "Word embedding based clustering to detect topics in social media," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell.*, Oct. 2019, pp. 192–199.
[15] F. Atefeh and W. Khreich, "A survey of techniques for event detection in Twitter," *Comput. Intell.*, vol. 31, no. 1, pp. 132–164, Feb. 2015.
[16] H.-T. Zheng, Z. Wang, W. Wang, A. K. Sangaiah, X. Xiao, and C. Zhao, "Learning-based topic detection using multiple features," *Concurrency Comput., Pract. Exper.*, vol. 30, no. 15, p. e4444, Aug. 2018.
[17] L. Ding, Y. Zhang, and J. Chen, "Hierarchical clustering for micro-learning units based on discovering cluster center by LDA," in *Proc. 9th Int. Conf. Inf. Technol. Med. Edu. (ITME)*, Oct. 2018, pp. 512–516.
[18] M. Manai, "A new approach for topic detection using adaptive neural networks," 2019, *arXiv:1903.03775*. [Online]. Available: http://arxiv.org/abs/1903.03775
[19] A. Dehghantanha, M. Conti, and T. Dargahi, "Cyber threat intelligence: Challenges and opportunities," *Cyber Threat Intell.*, vol. 70, pp. 1–6, Dec. 2018.
[20] P. Chen, N. L. Zhang, T. Liu, L. K. M. Poon, Z. Chen, and F. Khawar, "Latent tree models for hierarchical topic detection," *Artif. Intell.*, vol. 250, pp. 105–124, Sep. 2017.
[21] T. Joachims, "A probabilistic analysis of the rocchio algorithm with TFIDF for text categorization," Ph.D. dissertation, Dept Comput. Sci., Carnegie-Mellon Univ, Pittsburgh, PA, USA, 1996.
[22] S. Yu, J. Su, and P. Li, "Automatic Abstract Extraction Method Based on Improved TexRank," *Comput. Sci.*, vol. 43, 2016.
[23] X. Rong, "Word2vec parameter learning explained," 2014, *arXiv:1411.2738*. [Online]. Available: http://arxiv.org/abs/1411.2738
[24] J.-F. Yeh, Y.-S. Tan, and C.-H. Lee, "Topic detection and tracking for conversational content by using conceptual dynamic latent Dirichlet allocation," *Neurocomputing*, vol. 216, pp. 310–318, Dec. 2016.
[25] T. Yamada, "Detection of topics from newspaper and its analysis of temporal variations in regions," in *Proc. Pacific Neighborhood Consortium Annu. Conf. Joint Meetings (PNC)*, Nov. 2017, pp. 44–49.
[26] D. Yan, E. Hua, and B. Hu, "An improved single-pass algorithm for chinese microblog topic detection and tracking," in *Proc. IEEE Int. Congr. Big Data (BigData Congress)*, Jun. 2016, pp. 251–258.
[27] E. F. Tjong Kim Sang, "Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition," in *Proc. 6th Conf. Natural Lang. Learn.*, 2002, pp. 1–10.
[28] W. H. E. Day and H. Edelsbrunner, "Efficient algorithms for agglomerative hierarchical clustering methods," *J. Classification*, vol. 1, no. 1, pp. 7–24, Dec. 1984.
[29] F. Murtagh and P. Contreras, *Algorithms for hierarchical clustering: An overview II*, vol. 7. Hoboken, NJ, USA: Wiley, 2017.

XIAOFENG LU (Member, IEEE) received the Ph.D. degree from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2010. He held a Visiting Scholar position with the Computer Laboratory, University of Cambridge, U.K. He is currently an Associate Professor with the School of Cyberspace Security, Beijing University of Posts and Telecommunications. His main research interests include cyberspace security, information security, and artificial intelligence.

XIAO ZHOU received the bachelor's degree in computer science and technology from the Wuhan University of Engineering, China, in 2016, and the master's degree in computer science and technology from the Beijing University of Posts and Telecommunications, in 2019. Her main research interests include game artificial intelligence and deep learning.

WENTING WANG received the M.S. degree in electronic engineering from Shandong University, Jinan, China, in 2015. She is currently a Senior Engineer with State Grid Shandong Electric Power Company, Jinan. Her main research interests include electronic engineering and information security.

PIETRO LIÒ is currently a Professor with the Computer Laboratory, University of Cambridge, U.K., and a Fellow and the Director of studies with the Fitzwilliam College, University of Cambridge. He is also modeling biological processes on networks, modeling stem cells, and developing transcription and phylogenetic applications on a grid environment. He is also interested in bio-inspired design of wireless networks and epidemiological networks.

PAN HUI (Fellow, IEEE) received the bachelor's and M.Phil. degrees from The University of Hong Kong and the Ph.D. degree from the Computer Laboratory, University of Cambridge. He was affiliated with the Intel Research Cambridge. He is currently a Professor of computer science and engineering with The Hong Kong University of Science and Technology. He is also a Distinguished Scientist with Deutsche Telekom Laboratories (T-Labs), Berlin. His research interests include delay tolerant networking, mobile networking and systems, planet-scale mobility measurement, social networks, and the application of complex network science in communication system design.