Overview of clustering analysis algorithms in unknown protocol recognition

Zhiguo Liu*, *Changqing* Ren *and Wenzhu* Cai

Communication and Network Laboratory (Dalian University), No.10 xuefu street, Dalian economic and technological development zone, liaoning, Dalian 116622, China

Keywords: Unknown protocol recognition, Cluster analysis, Similarity measure, Bit stream.

Abstract. In the process of identifying unknown protocol of bit stream, the clustering of data sets of bit stream in the protocol is the basis of further identifying unknown protocol. Therefore, on the one hand, this paper analyses the classical clustering algorithms used in unknown protocol recognition from three perspectives: the whole process of clustering analysis, similarity measurement and clustering result evaluation. On the other hand, the development trend of clustering algorithm in unknown protocol recognition is summarized, and other problems in unknown protocol recognition can be solved by clustering algorithm according to the characteristics of bit stream data set, which can provide reference for future research work. Finally, the challenges faced by the Research Institute and the prospects for future work are given.

1 Introduction

In the field of network security and information confrontation, for the undisclosed communication protocol, mining the protocol format specification from the communication data has become a research hotspot of unknown protocol identification. Unknown protocol identification refers to the process of extracting protocol grammar, grammar and semantics by monitoring and analyzing the network input/output, system behavior and instruction execution flow of the protocol entity without relying on the protocol description [1].

In the fields of electronic countermeasures, the protocols used by both parties are customizable and non-public, and most of the intercepted communication data are continuous bit stream information. In the field of network supervision, the protocol parsing tools used in the network communication process also encounter many bit stream protocols that cannot be resolved. For these protocols, the protocol analyst does not have any prior knowledge, and it is very difficult to parse these completely unknown protocols [2]. Because we do not know any information related to the unknown protocol, such as format description, development document, etc., we can regard it as solving the unknown protocol identification problem under the condition of zero knowledge. In practical application environment, the captured

* Corresponding author: liuzhiguo_dldx@163.com

unknown protocol data frames are often mixed by multiple protocols. However, there are few studies on how to classify multi-protocols into single protocols under the condition of zero knowledge.

Cluster analysis is gradually developed along with the scientific development of statistics, computer science and artificial intelligence and other fields. Therefore, if there is great research progress in these fields, it will inevitably promote the rapid development of clustering analysis algorithms. So far, clustering technology has been widely used in many fields [3]. This paper mainly studies the application of clustering algorithm in data frame classification in unknown protocol identification. For the protocol data stream captured at the data link layer, which cannot be recognized by the capturer, the protocol development documents and protocol format descriptions are usually not disclosed. In order to further analyze such completely unknown protocols, the captured bit stream data frames need to be classified, that is, the unknown mixed multi-protocol data frames are classified into single-protocol data frames, which is an unsupervised classification process. Therefore, this paper mainly focuses on the clustering analysis algorithm in the unknown protocol bit stream clustering, and discusses the whole process of clustering analysis.

2 Cluster analysis process

Cluster analysis is a rigorous data analysis process. The whole process of cluster analysis is shown in Figure 1. From the data source to the evaluation of clustering results, it mainly includes three parts: feature selection or transformation, clustering algorithm selection, evaluation of clustering results, etc. [4]

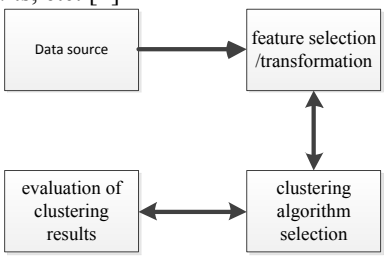

Fig. 1. Cluster analysis process diagram.

2.1 Feature selection or transformation

Generally, the sample data is chaotic. Clustering analysis first needs to deal with feature selection or transformation of data sets. In fact, feature selection and feature transformation are the two major categories of dimensionality reduction techniques. Feature selection refers to selecting from the all features (or attributes) of the data sample set a number of attributes that are more conducive to achieving a certain goal, that is, a subset of the original attribute set, and also achieve the purpose of reducing the dimension; Feature transformation refers to mapping the attributes of the original input space to a new feature space through some transformation, and then selecting some important transformed features according to the rules in the feature space. Based on the statistical characteristics of bit stream, Wang ZhaoFeng et al [5]. proposed three protocol irrelevant bit stream characteristic parameters, namely compression rate, hamming weight and run-length frequency, to complete the feature selection of the unknown protocol bit stream data set. Yue et al [6] used AC algorithm to mine frequent sequence features in binary data frames, and innovatively used Apriori algorithm to search and analyze the association relationship of these features. Yan Xiaoyong

et al [7] proposed a feature dimension reduction algorithm based on frequent items to reduce the dimension of binary protocol session flow because of the lack of features and the difficulty of extracting frequent patterns. The feature vectors were constructed from frequent items in protocol data to represent the original data frames. Liu et al [8] carried out principal component analysis (PCA) on the data set, and then selected the principal component to construct a new data set based on the analysis results. This algorithm reduces the dimension of the data set, greatly reduces the amount of calculation, and improves the efficiency of the algorithm. Feature selection or transformation plays an extremely important role in the process of clustering analysis, and the quality of the results will directly affect the final clustering results.

2.2 Clustering algorithms selection or design

The second part of clustering analysis is to select or design clustering algorithm according to the characteristics of data set after feature selection or transformation. If the sample set data are all numerical data, we need to pay attention to different dimensions when selecting or designing clustering algorithm. In general, sample set data is not necessarily numeric data. Therefore, clustering algorithms need to have the ability to process non-numeric data. The similarity measure between each sample point is the primary problem in clustering algorithm. Common methods of similarity metrics are described later.

The similarity measure has similar meaning to the "distance" between the samples mentioned frequently, but their values are diametrically opposite, that is, the larger the similarity measure, the closer the "distance" is. Wang Zhaofeng [5] et al proposed an initial clustering center selection method based on distance accumulation sum for the sensitivity problem of K-means algorithm to the initial clustering center by the proposed three uncorrelated bit stream characteristic parameters. The actual collected bit stream data set is clustered by k-means algorithm. The feature parameters defined by the method can be effectively used for clustering of unknown protocol bit streams, which can improve the stability and execution efficiency of the k-means algorithm. Yue [6] et al used the AC algorithm to mine the frequent sequence features in binary data frames, and then searched and analyzed the association relationship of these features through Apriori algorithm, combined with its characteristics to carry out four-step pruning processing, and finally passed the K-means algorithm. Perform clustering. Yan Xiaoyong [7] et al proposed a feature reduction algorithm based on frequent items and a density peak clustering algorithm based on distance index weighting, which can automatically select cluster centers, which effectively improves the discrimination between cluster centers and other data frames. However, the limitation of this algorithm is that it is not accurate enough to extract frequent items of different types of protocol data in the data set. Based on the traditional AGNES (AGglomerative NESting) hierarchical clustering algorithm, combined with the characteristics of bit stream data frames, Zhang FengLi [9] et al proposed a protocol classification algorithm based on improved condensed hierarchical clustering. The algorithm can automatically determine the number of clusters, which has stronger practical application value, but it is more likely to generate wrong clusters when dealing with data frames with lower similarity.

2.3 Evaluation of clustering results

Cluster can only be obtained by clustering termination criterion function. It should be pointed out that this criterion function is usually realized by artificial termination conditions, which do not have a unified criterion. It can be seen that cluster analysis is a supervised classification process, so after cluster generation, it is necessary to evaluate the clustering results comprehensively. The original goal of cluster analysis is to obtain data structures that are

implicit in a particular data set. What's more, for the same data set, different clustering algorithms generally get different clusters. However, in the cluster analysis of the unknown protocol bit stream, the clustering result can be compared with the data set used in the simulation experiment, and the clustering result is also relatively unique. In literature [10], in order to verify the performance of the algorithm, select *purity* and F use as evaluation indicators, such as (1) and (2).

purity :

$$purity = \sum_{c=1}^{z} \frac{\max(n_c^t)}{n} \tag{1}$$

$$t \in \{1,2,3,\cdots,r\}$$

where n is the number of protocol data frames, z is the number of clusters, r is the actual number of classes, and n_c^t is the number of data frames belonging to class t in cluster c.

F :

$$recall(t,c) = \frac{n_c^t}{n^t} \tag{2}$$

$$precision(t,c) = \frac{n_c^t}{n_c}$$

$recall$ and $precision$ are the recall rate and accuracy, respectively. n^t is the number of data frames belonging to class t . n_c is the number of data frames belonging to cluster c .

$$F(t,c) = \frac{2 \times recall(t,c) \times precision(t,c)}{recall(t,c) + precision(t,c)} \tag{3}$$

$F(t,c)$ is the F value between the class t and the cluster c .

$$F = \sum_{t=1}^{r} \frac{n_t}{n} \max\{F(t,c)\} \tag{4}$$

3 Similarity measure

Cluster analysis is a method of classifying similarity samples of data sets into several classes. Therefore, how to measure the similarity between samples is a key issue in clustering algorithms. Assuming that the similarity between samples satisfies symmetry, non-negativity and reflexivity, the similarity between samples is called Metric. The characteristics of data sets are generally divided into three types: continuous variables (or quantitative variables), discrete variables (or qualitative variables), and mixed variables. In the data set of the unknown protocol bit stream data frame, its characteristics are generally continuous variables.

3.1 Euclidean distance

This is one of the most commonly used methods for measuring the distance between samples. The calculation formula is as follows:

$$D(X_i, X_j) = \sqrt{\sum_{l=1}^{d} (X_{il}, X_{jl})^2} \tag{5}$$

Here, D denotes the distance between samples; l denotes the dimension of sample features; d denotes the total dimension of samples (the same below), that is, the total number of sample features. Euclidean distance is a two-norm form with invariance of transformation and rotation in feature space, and generally tends to construct spherical clusters. Then, a large difference in the attribute values or a linear transformation will cause the correlation to be deformed [11,12].

In order to solve this problem, it is necessary to standardize the target data set so that each attribute contributes the same rate to distance. This is also the conventional way to eliminate dimensional differences between features. Before the data analysis, the sample set needs to be standardized on the mean and variance [13]. The standardized calculation formula is as follows:

$$x_{il} = \frac{x_{il}^* - m_l}{S_l} \tag{6}$$

Where m is the mean; S is the variance; * represents the original value of the feature (the same below). In addition, in order to remove the difference in dimension between different attribute values, it is necessary to regularize the sample set. For example, the regularization formula in the interval [0,1] is:

$$x_{il} = \frac{x_{il}^* - \min(x_{il}^*)}{\max(x_{il}^*) - \min(x_{il}^*)} \tag{7}$$

3.2 Chebyshev distance

In two-dimensional space, the typical application of the Chebyshev distance is to solve the problem that the king in chess moves from one grid to another at least a few steps. This distance has been effectively applied in the fuzzy C-MEANS method. The formula for Chebyshev distance can be expressed as:

$$D(X_i, X_j) = \max_l(\left|X_{il} - X_{jl}\right|) \tag{8}$$

Another representation of this formula is:

$$D(X_i, X_j) = \lim_{p\to\infty} \sqrt[p]{\sum_{l=1}^{d} (X_{il} - X_{jl})^2} \tag{9}$$

3.3 Point symmetry distance

When there are symmetric patterns in clustering, the distance between symmetric points can be used. Its expression formula is as follows:

$$D(X_i, X_j) = \max_{\substack{j=1,\cdots,N \\ and\ j\neq i}} \frac{\left\|(X_i - X_r) + (X_j - X_r)\right\|}{\left\|(X_i - X_r)\right\| + \left\|(X_j - X_r)\right\|} \tag{10}$$

The distance from the symmetric point is the minimum distance from the point to the symmetric point and other points.

3.4 Correlation coefficient

Distance measures can also be derived from correlation coefficients, such as Pearson correlation coefficients defined as:

$$\rho_{X_i X_j} = \frac{Cov(X_i, X_j)}{\sqrt{D(X_i)}\sqrt{D(X_j)}} \tag{11}$$

3.5 Cosine similarity

Cosine similarity is a direct method to calculate similarity. Its form of expression is:

$$S(X_i, X_j) = \cos\alpha = \frac{X_i^T X_j}{\left\|X_i\right\|\left\|X_j\right\|} \tag{12}$$

Here, S represents the similarity between samples (the same below). In the feature space, the more similar the two samples are, the more they tend to be parallel, and the greater their cosine values.

In the clustering of unknown protocol bit stream data frames, the common similarity measure method based on hierarchical clustering algorithm is shown in Figure 2. There are generally four methods for measuring the distance between clusters:

3.6 Single-link

One data point p and p', C_A and C_B are selected from cluster A and B to represent the set of points of cluster A and B respectively. The distances of all such point pairs are calculated, and the distances $D_{\min}(C_A, C_B)$ of the two clusters are expressed by the nearest two points. The calculation formula is as follows:

$$D_{\min}(C_A, C_B) = \min_{p\in C_A, P'\in C_B} \left|p - p'\right| \tag{13}$$

3.7 Complete-link

In the same way, the distance $D_{\max}(C_A, C_B)$ of the two clusters is expressed by two points farthest from each other. The calculation formula is as follows:

$$D_{\min}(C_A, C_B) = \min_{p \in C_A, P' \in C_B} |p - p'| \tag{14}$$

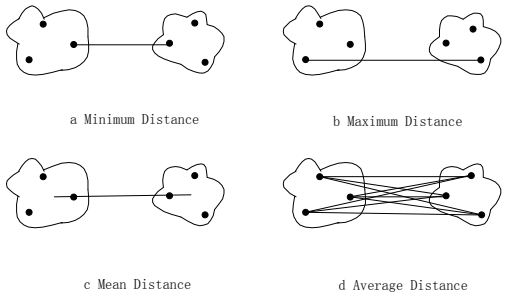

Fig. 2. Schematic diagram of various distance measurement methods.

3.8 Mean-link

In the same way, the distance $D_{mean}(C_A, C_B)$ between the two clusters is expressed by the distance between the two central points. The calculation formula is as follows:

$$D_{mean}(C_A, C_B) = |m_A - m_B| \tag{15}$$

where m_A is the average of cluster C_A and m_B is the average of cluster C_B.

3.9 Average-link

One data point p and p' are selected from cluster A and cluster B, and the distances of all such point pairs are calculated, and their average values are calculated. The obtained average values are used to represent the distances $D_{avg}(C_A, C_B)$ of the two clusters.

$$D_{avg}(C_A, C_B) = \frac{1}{n_A n_B} \sum_{p \in C_A} \sum_{p' \in C_B} |p - p'| \tag{16}$$

Where n_A is the number of data points in C_A, and N_B is the number of data points in C_B. In reference [10], for the classification of unknown protocols, the input data is a mixed multi-protocol data frame. Two methods are used to define the similarity between data frames i and j: One is the similarity between data frames customized by the protocol data frame feature; The other is the definition of string similarity based on edit distance calculation method, which is suitable for general situations. They are described as follows: The similarity between data frames (sequences) is represented by i and j, and $similar1(i, j)$ by the similarity between the two data frames. Then the calculation formula is defined as follows:

$$similar1(i, j) = \frac{sam(i, j)}{sum(i, j)} \tag{17}$$

Where, $sam(i, j)$ is the result obtained by following the following operations; The sequences i and j are aligned to the left end, in units of 4 bits, and compared from left to

right, the characters are the same plus one, and the difference is not added. $sum(i, j)$ is the number of comparisons when calculating $sam(i, j)$. It is usually the smaller of the length of i and j sequences. The minimum similarity between data frames is 0 and the maximum is 1.

String similarity is to treat two data frames (sequences) as two strings. Then the formula for calculating the similarity $similar2(i, j)$ between sequences is as follows:

$$similar2(i, j) = 1 - \frac{Dis\tan ce(i, j)}{\max(length(i), length(j))} \tag{18}$$

Where, i and j are two data frames (strings), $length(i)$ and $length(j)$ are the lengths of data frames (strings) i and j respectively. $Dis\tan ce(i, j)$ is the editing distance of two strings, and represents the minimum number of operations required to turn string i into string j through "insert", "replace", "delete" and other operations.

Whether the similarity measurement method between samples is reasonable will directly affect the final clustering effect, which is particularly important. For a specific sample set, which similarity calculation method is the most appropriate? How to explain the physical meaning of similarity? There is no definitive answer to these questions so far. Therefore, the cluster analysis method is inevitably characterized by subjectivity and dependence on the problem domain.

4 Research on clustering algorithm

So far, clustering research and its application fields have been very extensive. In the process of bit stream unknown protocol identification, clustering is to separate binary data frames acquired in a complex wireless network environment, and the obtained multi-protocol data frames are divided into single-protocol data frames, which provides preconditions for subsequent protocol reverse analysis. This paper discusses common clustering methods for data frame classification applied in the process of unknown protocol identification.

4.1 K-means

K-means algorithm is a widely used clustering algorithm [14,15]. Its core idea is that the user specifies each initial center (random number) of k as the category of the cluster, and iterates repeatedly until the algorithm converges. First, calculate the distance from all data points to the k initial center, and use this calculated distance as the next classification criterion. That is to say, the nearest distance from each data point to that center determines which category it belongs to in this classification. Then, the initial defined k center will divide all the data into k categories in the iteration, that is, k clusters. After the distance calculation is performed for each sample point and the category is assigned, the center corresponding to each cluster of the k clusters is recalculated, that is, the center is updated. Each cluster data is clear, the center can be actually obtained, and then the center is replaced by the center as a new distance calculation standard, and the process of repeating the distance is a cluster. After that, the center update and data clustering process is repeated until the center is updated, and the center of each cluster does not change or only slightly changes, the algorithm stops. The resulting k final center and the sample points contained in their clusters are the desired clustering results.

The value of k in the K-means algorithm is also the number of categories of clusters that need to be defined by the user. When encountering a complex data structure, it may take

multiple attempts to select a better k value to make the sample data gather. It is optimal to have so many classes.

It can be found that the initial center selected in the initial algorithm iteration is randomly defined, which will result in poor clustering effect and increased number of iterations, and may only obtain local optimal results. Local optimization is a common problem of k-means algorithm. Meanwhile, k-means algorithm is only applicable to data clustering, and when noise data appear, due to the principle of the algorithm, the sum of squares of distances is taken as the criterion, which will make some unreasonable extreme data affect the clustering results.

To overcome these shortcomings, in [9], the user specifies the initial center, which translates the condition of "user specifies the number of clusters k" into the rule of "user specifies the satisfactory initial center ". For users, the improved condition is easier to achieve; In addition, the user can set the number of satisfactory objects in the cluster to achieve the clustering edge extraction result and find a better cluster in time. This method can increase the scalability of the algorithm to some extent. in [16], A silhouette coefficient is introduced into the hierarchical clustering to confirm the optimal clustering number of binary frames. The iterative hierarchical clustering based on silhouette coefficient is exploited to keep the precision and reduce the time-space complexity of the hierarchical clustering. Each subspace includes different clustering sizes to provide the diversity of frames for next algorithm called improved multiple sequence alignment algorithm. Furthermore, the accuracy and efficiency of unknown protocol recognition are improved.

4.2 DBSCAN

DBSCAN is a density-based clustering algorithm [17,18]. Widely used in various fields to identify clusters of arbitrary shapes. Its clustering idea is to derive the maximum density-connected sample set available in density-receiving relationship. This is a clustered cluster.

The basic implementation steps are as follows:

- determine neighborhood parameters (ε , *MinPts*);
- initialize core object sample set Ω , cluster number k, cluster partition C and unvisited sample set δ ;
- Find out all the core objects:

(1) For any sample, the neighborhood is the set of subsamples in which all the samples are not more than ε from the sample, then the neighborhood subsample set is found.

(2) Judge the subsample set: if the number of samples in the neighborhood subsample set of a sample is no less than *MinPts* , the sample is a core object, and add it to the core object sample set Ω .

- select a core object X_C randomly from the core object sample set Ω obtained by us, initialize the core object sample of the current cluster, so that the core object sample set at this time only has the core object sample selected randomly, and call the core object sample set after initialization as Ω^* . that is $\Omega^* = \{X_C\}$. And initializes the class ordinal number $k=k+1$, initializes the current cluster sample set $C_k = \{X_C\}$, and updates the sample set $\delta = \delta - X_C$.
- Take a core object in the current cluster core object sample set Ω^*, then take X_C for the first time. Taking X_C as an example, first find the neighbor subsample set $N_\varepsilon(X_C)$ by distance constraint, let $\beta=N_\varepsilon(X_C)\cap\delta$, here we have to consider the intersection of different core object samples, and the intersection of the neighborhood and the unvisited sample set means to remove this. A sample that has been processed in a neighborhood. Then

the resulting set β is placed in the current cluster sample set, that is, $C_k=C_k\cup\beta$ is updated, then the unvisited set naturally has less sample elements, and needs to be updated, and then updated to $\delta=\delta-\beta$. It is not known whether other core sample objects are included in the neighborhood subsample set. That is to say, when we put the subsample set into the current cluster sample set, other core objects may be entered together, and then these subsample sets are clustered. The core objects are not added to the current cluster core object collection Ω^* for processing, we need to find them back to join them. Then update $\Omega^*=\Omega^*\cup(\beta\cap\Omega)-X_C$.

- Repeat the last two steps with another core object.

DBSCAN is applicable to any dense data and is not sensitive to noise data in the data. The clustering result is less interfered by other factors, which can avoid the situation that K-means is greatly affected by the initial value. However, due to its density-based algorithm characteristics, if the sample density is uneven, the clustering quality will be low. Moreover, if there are many sample data, DBSCAN clustering will take a long time to converge, which is not ideal.

Reference addresses the user-defined problem of parameter dependence in DBSCAN. A method for combining the information of k-nearest neighbor is used with DBSCAN to achieve a parameter-free clustering technique. The KNN method for finding kernels assumes the entire dataset to be a directed graph. Every data object is a node in a graph. Find the strongly connected component in the data graph as the threshold *MinPts*, Calculate the maximum value of all third nearest neighbor distances of all points as the threshold *Eps*. Expand the cluster with DBSCAN 's density accessibility, then output and repeat the process until the clustering is complete, thus achieving parameter-free clustering. In [20], in order to solve the problem of *Eps* and *MinPts* parameter selection of DBSCAN clustering algorithm, a neighborhoods-independent dynamic parameter selection method is proposed. Firstly, the data set is preliminarily clustered based on the k-means algorithm. For the result of the preliminary clustering, the distance value with the largest sample logarithm is selected as the *Eps* value of the corresponding class, and then the *MinPts* value is obtained by *Eps* ,and finally the DBSCAN algorithm is selected. The improvement is made to adaptively adjust the running value according to the *Eps* corresponding to the k-means cluster of the current core point. The results show that this idea can be applied to bit stream clustering analysis under the condition of unknown protocol , and satisfactory clustering results can be obtained without users specifying *Eps* and *MinPts* , which improves the applicability and accuracy of the algorithm.

4.3 Hierarchical clustering algorithm

Hierarchical clustering includes bottom-up agglomerative clustering and top-down split clustering methods [21]. Aggregated clustering requires that we first treat each data point as a single cluster, and then merge the two nearest clusters. After the merger, the distance between clusters is recalculated and the merger steps are repeated until the clustering is a cluster or the default conditions are reached.

Split clustering is the opposite of agglomerated clustering, which aims to split a large data set into many small clusters. First, all objects are placed in a class, and then gradually divide into small clusters, so that the total data set becomes smaller and smaller but more and more clusters, until each object becomes a cluster alone or meets certain termination conditions.

Hierarchical clustering does not require us to get the number of clusters that need to be synthesized or segmented in advance. Meanwhile, it can use a variety of distance calculation methods to calculate the distance between clusters. However, due to the large amount of distance calculation, hierarchical clustering is not fast and inefficient [22]. In [9], in the

traditional agglomerative hierarchical clustering algorithm, combined with the characteristics of the bit stream data frame, the similarity between data frames and clusters is defined, and the number of clusters can be automatically determined. And clustering data frames quickly and efficiently. In reference [23], in order to solve the problem of unknown protocol data frame classification, hierarchical clustering is selected as the protocol frame classification method, and the optimal cluster number is determined by introducing contour coefficient, so that the set of single protocol data frames used in the next algorithm can be accurately obtained.

5 Development trend of clustering algorithm in unknown identification

In the complex network environment where multiple users, multiple applications and multiple protocols coexist, it is the basis for further protocol identification and analysis to classify bit streams with similar protocol properties into corresponding categories. The first challenge of unknown protocol identification is whether it can accurately classify and further identify the protocol information and user data contained in it without any prior knowledge, and the definition and extraction of bit stream feature parameters and based on these Clustering of feature parameters is the basis for carrying out the identification and analysis of unknown protocols. In order to solve this problem, many researchers have studied and improved k-means, AGNES, DBSCAN and other classical algorithms according to the characteristics of binary data frames, and gradually formed a more perfect clustering strategy. However, when dealing with the classification of binary mixed data frames, wrong clustering still occurs. And it affects the whole process of unknown protocol identification. This undoubtedly puts higher requirements on the clustering algorithm. In 2014, Italian researcher Rodriguez et al. proposed a new efficient clustering algorithm: Fast density peaks clustering [24], The advantages of this algorithm are that the number of clusters can be generated intuitively, the outliers can be found and eliminated automatically, and the clustering can be completed by any shape and any dimension of mapping space. In literature [10], this algorithm is applied to the recognition of unknown protocols, which greatly improves the recognition accuracy. However, as this algorithm needs to calculate the distance between any two samples in advance, the similarity calculation of this algorithm costs a lot [25]. In the unknown protocol recognition, a better clustering algorithm is needed to solve the problem of data frame classification. For the current research, the similarity measures in various clustering algorithms can be combined to improve the performance of the clustering algorithm. Thereby improving the efficiency of the entire unknown protocol identification.

6 Conclusion

There are many specific algorithms for cluster analysis, which can be used to get the clustering results of samples. So far, many clustering algorithms based on various ideas and theoretical basis have been proposed, and the practical application of the algorithms is becoming more and more mature. However, practice has proved that no clustering algorithm can be universally applicable to reveal various structures presented by various cubes, and no clustering algorithm is superior to other algorithms for all data types and definition domains. Each relatively superior algorithm has specific application environment. In terms of the application of clustering analysis algorithm in the recognition of unknown protocols, one or more similarity measures should be adopted to improve and optimize the clustering algorithm according to the characteristics of binary protocol data frames, so as to improve the accuracy of the whole unknown protocol recognition.

In addition, in the process of classifying unknown data frame identification data, many researchers are always answering the question of different protocols being classified. However, in addition to the mixed features of unclear protocols, in addition to answering the application of clustering analysis algorithms in the classification of different protocols, it is also necessary to study the different formats of the cluster analysis algorithm in the same protocol and the difference in the same format. The application in the attribute classification.

This research is supported by the General Project from Field Fund of Equipment Development Ministry (No. 61403110308); the National Defense Research Foundation of China (No. 61405180402).

References

1. F. Pan, L. F. Wu, Y. X. Du, Z. Hong, Overviews on Protocol Reverse Engineering, Journal of Computer Application Research, **28**(2011)2801-2806.
2. X. D. Li, L. Chen, A Survey on Methods of Automatic Protocol Reverse Engineering, in: Seventh International Conference on Computational Intelligence and Security (CIS 2011), Hainan, (2011), pp. 685-689.
3. C. C. Zhang, H. Y. Zhang, Improved K-means Algorithm Based on Density Canopy, Knowledge-Based Systems, **145**(2018)289-297.
4. Y. L. Zhang, Y. J. Zhou, Review of Clustering Algorithms, Journal of Computer Applications, **39**(2019)1869-1882.
5. Z. F. Wang, Y. Wu, Unknown Protocol Bit Stream Clustering Based on Improved k-means Algorithm, Journal of Computer Applications, **36**(2016)5-8.
6. Y. Yue, F. Z. Men, C. R. Zhang, T. Li, Cluster System for Binary Data Frame, Application Research of Computers, **32**(2015)909-911+916.
7. X. Y. Yan, Q. Li, Clustering Algorithm for Binary Protocol Data Frames Combining Feature Dimensionality Reduction and Density Peaks Clustering, Journal of Chinese Computer Systems, **39**(2018)2662-2668.
8. N. Liu, J. H. Si, The Analysis Research of Clustering Algorithm Based on PCA, in: 2017 13th IEEE International Conference on Electronic Measurement and Instruments (ICEMI), Yangzhou, (2017), pp. 361-365.
9. F. L. Zhang, H. C. Zhou, J. J. Zhang, Y. Liu, C. R. Zhang, A Protocol Classification Algorithm Based on Improved AGNES, Computer Engineering and Science, **39**(2017)796-803.
10. X. Y. Yan, Q. Li, Y. T. Si, A Clustering Algorithm for Binary Protocol Data Frames Based on Principal Component Analysis and Density Peaks Clustering, in: 2017 IEEE 17th International Conference on Communication Technology (ICCT), Chengdu, (2017), pp. 1260-1266.
11. R. O. Dud, P. E. Hart, D. G. Stork, Pattern Classification (2nd Edition), John Wiley and Sons., Inc., New York, (2001).
12. Gao, Y. K. Wang, J. Li, Bounds on Covering Radius of Linear Codes with Chinese Euclidean Distance over the Finite Non Chain Ring F2+vF2, Information Processing Letters, **138**(2018)22-26.
13. R. Hogg, E. Tanis, Probability and Statistical Inference (7 edition), Prentice Hall., Inc., Upper Saddle River, (2005).
14. J. P. Qi, Y. W. Yu, L. H. Wang, J. L. Liu, K*-Means: An Effective and Efficient K-Means Clustering Algorithm, in: 2016 IEEE International Conferences on Big Data and

Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom) (BDCloud-SocialCom-SustainCom), Atlanta, GA, (2016), pp. 242-249.

15. H. B. Shi, M. Xu, A Data Classification Method Using Genetic Algorithm and K-Means Algorithm with Optimizing Initial Cluster Center, in: 2018 IEEE International Conference on Computer and Communication Engineering Technology (CCET), Beijing, (2018), pp. 224-228.
16. S. Y. Tao, H. Y. Yu, Q. Li, Bit-oriented Format Extraction Approach for Automatic Binary Protocol Reverse Engineering, in: IET Communications, **10**(2016)709-716.
17. M. A. Li, D. X. Meng, S. Y. Gu, S. F. Liu, Research and Improvement of DBSCAN Cluster Algorithm, in: 2015 7th International Conference on Information Technology in Medicine and Education (ITME), Huangshan, (2015) pp. 537-540.
18. L. Zhu, J. Zhu, C. M. Bao, L. H. Zhou, C. Y. Wang, B. Kong, Improvement of DBSCAN Algorithm Based on Adaptive Eps Parameter Estimation, in: Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence, Sanya, (2018), pp. 27:1-27:7.
19. A. Sharma, A. Sharma, KNN-DBSCAN: Using K-Nearest Neighbor Information for Parameter-Free Density Based Clustering, in: 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kannur, (2017), pp. 787-792.
20. Z. F. Wang, G. L. Shan, Characteristic Parameters Extraction and Correlation Analysis of Unknown Protocol Bit Streams, in: 2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC), Qinhuangdao, (2015), pp. 1502-1505.
21. Z. Nazari, D. Kang, M. R. Asharif, Y. Sung S. Ogawa, A new hierarchical clustering algorithm, in: 2015 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), Okinawa, (2015), pp. 148-152.
22. J. Mohammad, S. E. Faramarz, B. Zahra, combining hierarchical clustering approaches using the PCA method, Expert Systems with Applications,**137**(2019)1-10.
23. F. Z. Meng, C. R. Zhang, G. Wu, Protocol reverse based on hierarchical clustering and probability alignment from network traces, in: 2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA), Shanghai, (2019), pp. 443-447.
24. A. Rodriguez, A. Laio, Clustering by Fast Search and Find of Density Peaks, Science, **344**(2014)1492-1496.
25. X. Xu, S. F. Ding, T. F. Sun, A Fast Density Peaks Clustering Algorithm Based on Pre-Screening, in: 2018 IEEE International Conference on Big Data and Smart Computing (BigComp), Shanghai, (2018), pp. 513-516.