Research Article

Controllable Clustering Algorithm for Associated Real-Time Streaming Big Data Based on Multi-Source Data Fusion

Haiting Cui and Shanshan Li

School of Sport Communication and Information Technology, Shandong Sport University, Jinan 250102, China

Correspondence should be addressed to Haiting Cui; cuihaiting@sdpei.edu.cn

Received 2 November 2021; Revised 19 January 2022; Accepted 21 January 2022; Published 23 February 2022

Academic Editor: Narasimhan Venkateswaran

Copyright © 2022 Haiting Cui and Shanshan Li. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the problems of poor security and clustering accuracy in current data clustering algorithms, a controllable clustering algorithm for real-time streaming big data based on multi-source data fusion is proposed. The FIR filter structure model is used to suppress network interference, and ant colony algorithm is used to detect the abnormal data in the big data. By optimizing the iteration, the pheromone concentration is placed in the front position as the abnormal data point, and the filter is introduced. The fusion scope of multi-source data fusion is set. Combined with the data similarity function, the multi-source data fusion concept is used to construct the associated real-time streaming big data fusion device, and the data deduplication results are substituted into the fusion device to obtain the data clustering result. The experiments show that the proposed algorithm has high safety factor, good data clustering accuracy, high clustering efficiency, and low energy consumption.

1. Introduction

The arrival of big data not only promotes industrial development but also promotes the evolution of new business models [1, 2]. To this end, the rapid mining and discovery of knowledge from massive data has become the focus of researchers and companies. Data clustering is an important research direction in data mining. It aims to divide data into categories consisting of similar objects by clustering data [3, 4]. However, the association of real-time streaming big data is versatile and complex. Traditional frameworks only consider computing resources for data stream extraction and storage and cannot effectively use computing resources to provide more and faster clustering services [5–7]. This requires a new data clustering algorithm [8, 9].

Cao and Qian proposed a big data clustering algorithm based on local key nodes. For the initial node uncertainty and the time consumption caused by the fitness function calculation, local key nodes were introduced and the fitness formula was improved to reduce the time consumption. Experiments with classical algorithms in small-scale data networks and large-scale data networks showed that the

clustering time was short, but there was a problem of poor security [10–13]. Zhang et al. proposed a SOM hybrid attribute data clustering algorithm based on heterogeneous value difference metrics. The algorithm used self-organizing map neural network as the framework and used heterogeneity difference based on sample probability to measure the dissimilarity of mixed attribute data. The frequency of occurrence of the classification feature in the Voronoi set was used as the basis of the reference vector update rule of classification attribute data, and the update of the numerical attribute and the classification attribute data rule was realized by the hybrid update rule. The proposed clustering algorithm was tested by using the classification attribute and mixed attribute dataset in UCI public database. The experimental results showed that the algorithm had low running complexity, but there was a problem of poor classification accuracy [14]. Wang et al. proposed a data classification algorithm based on Spark-FCM. Firstly, the matrix was distributed by horizontal partitioning, and different vectors were stored in different nodes. Then, based on the computational characteristics of FCM algorithm, a distributed and cache sensitive common matrix operation

and Spark-FCM algorithm are designed. The main data structure adopted distributed matrix storage, which had less data movement between nodes and distributed computing features at each step. The experimental results showed that the algorithm had good stability, but it had a long time-consuming problem [15–18]. Liu et al. proposed an improved manifold clustering algorithm based on density peak search. The global and local spatial manifold distribution of the dataset was comprehensively considered, and the local density of each sample point was defined. According to the local density of each sample point and its local density relationship with other sample points, the cluster center criterion was defined. Clustering is implemented. But the algorithm had certain classification accuracy, and there was a problem of high energy consumption of clusters [19].

Aiming at the problems existing in the current research results, a controllable clustering algorithm for associated real-time streaming big data based on multi-source data fusion is proposed. The general framework is as follows:

- (1) The anti-interference filtering of the associated real-time streaming big data is realized by FIR filtering algorithm, and the abnormal data in the filtering result by ant colony algorithm are detected and filtered out to improve the data clustering security in real time.
- (2) Redundant data in associated real-time streaming big data are removed to reduce energy consumption of data clustering.
- (3) Data clustering is achieved through multi-source data fusion.
- (4) The experiment and discussion method are used to verify the controllable clustering algorithm for associated real-time streaming big data based on multi-source data fusion.
- (5) The full text is summarized and the next research direction is proposed.

2. Methods

2.1. Processing Abnormal Associated Real-Time Streaming Big Data. In order to improve the security of the data clustering process, the abnormal data need to be eliminated [20–23]. In the process, the FIR filtering algorithm is first used to realize anti-interference filtering of the associated real-time streaming big data. The structure diagram is shown in Figure 1. Assuming that the data traffic is generated by a linearly correlated nonlinear time series, the following FIR filter structure model is used for interference suppression:

$$x_n = a_0 + \sum_{i=1} a_i x_{n-i} + \sum_{j=0} b_j \eta_{n-j}, \quad (1)$$

where x_n represents the network traffic information model of data center, a_0 represents the sampling amplitude of the initial network traffic, x_{n-i} represents the scalar time series of network traffic with the same mean and variance in data center, b_j represents the oscillation amplitude of the network traffic in data center, and η_n represents the delay sequence of data transmission.

According to the calculation of equation (1), the network traffic information flow of data center is Fourier transformed to obtain the time series $x(k)$, and the oscillation attenuation of the network traffic is obtained after the interference filtering process:

$$A(t) = \sqrt{am} B^H(t) \cdot x_n, \quad (2)$$

where a represents the inter-domain variance coefficient of network traffic, m represents the embedded dimension in phase space, and $B^H(t)$ represents the correlation function of data flow anomaly feature detection. Assuming that the input sequence $x(k)$ is a set of wide stationary time series, the transfer function $H_B(c)$ of the filter is

$$H_B(c) = \frac{(1 + \sin \theta_2)}{\cos \theta_2} \frac{\cos \theta_1(k) \cos \theta_2 c^{-1}}{1 + \sin \theta_1(k) (1 + \sin \theta_2) c^{-1} + \sin \theta_2 c^{-2}} G(c), \quad (3)$$

where

$$G(c) = \frac{1 - \sin \theta_2}{2} \frac{1 - c^2}{1 + \sin \theta_1(k) (1 + \sin \theta_2) c^{-1} + \sin \theta_2 c^{-2}}, \quad (4)$$

where $G(c)$ represents the filter transfer model.

Based on the above calculations, the designed interference suppression FIR filter of network traffic is as follows.

According to the above calculation and analysis, the final data streaming output result $c(t)$ obtained by the FIR anti-interference filtering process is

$$c(t) = x(t) + y(t) = A(t) + n(t), \quad (5)$$

where $x(t)$ represents the real part of the data streaming time series, $y(t)$ represents the imaginary part of the data streaming time series, and $n(t)$ represents the other influence vectors.

Based on the above interference suppression result, the ant colony algorithm is used to detect the abnormal data therein. The testing process is mainly divided into the following aspects:

FIGURE 1: Structure of FIR filter.

- (1) Set ant colony parameters.
- (2) Initialize the ant to find the abnormal data path pheromone value and create an empty table S .
- (3) Evenly distribute the same number of ants on the head node of the dataset graph DG .
- (4) Each ant selects a path L on DG according to the path selection rule.
- (5) Update the pheromone of path L .
- (6) Return to step (4) until convergence or meet the termination condition.
- (7) Select all paths whose pheromone is greater than the set threshold and save or modify S as required. The determination of the pheromone concentration in the table in the front position is determined as abnormal data.

In step (2), the pheromone $\chi_{i'}$ of each edge $e_{i'}$ ($1 \leq i' \leq En$) on DG is expressed as follows:

$$c_{ilc}(t) = \frac{1}{En}, \quad (6)$$

where En represents the number of edges in DG .

In step (2), the table $S = (T, A, V, M)$, where T represents the tuple address of the ant walking the path, A is the target attribute name, V is the target attribute value in the tuple, and M is the attribute measure value of A in the tuple.

In step (4), the node v is given, and the probability $P_{i'}(t)$ of the ant selecting the adjacent edge $e_{i'}$ is expressed as follows:

$$P_{i'}(t) = \frac{\chi_{i'}(t)\lambda_{i'}}{\sum \chi_{i'}\lambda_{i'}}, \quad (7)$$

where $\lambda_{i'}$ represents the edge heuristic factor. The larger the value is, the greater the probability of selecting this path is.

In step (5), the pheromone update on each side of the most recent path L is expressed as follows:

$$\chi_{i'}(t+1) = (1-\rho)\chi_{i'}(t) + \frac{\chi_{i'}(t)}{Q}, \quad (8)$$

where ρ represents the volatilization rate of pheromone, which can effectively inhibit the ant colony from rapidly converging to the path that has already passed.

According to the continuous update of the pheromone, the pheromone whose concentration exceeds the set threshold w is stored in the table S , and the abnormal data are judged. The abnormal data filter is introduced here to filter out the abnormal data, and the obtained normal data set can be expressed as

$$C(t) = F(x) \cdot \frac{c(t)}{\chi_{i'} \odot w}, \quad (9)$$

where $C(t)$ represents the normal dataset in the associated real-time streaming big data and $F(x)$ represents the abnormal data filter.

2.2. Deduplication of Real-Time Streaming Big Data. In order to reduce the burden of data clustering, reduce the energy consumption of data clustering, and improve the classification accuracy, the redundant data need to be cleared [24]. For a new data segment, there are two similarities in its similar characteristics, as shown by the black points A_1 and A_2 in Figure 2, respectively, wherein, the black dot A_1 indicates that the position of the data segment in the plane is outside the three class boundaries, and the most similar data segment is the closest to the distance, and the white points are at the boundary of the class [25, 26]. The black dot A_2 indicates that the position of the data segment in the plane is inside the class, and the most similar data segment is the closest distance between several white dots and black dots in the same class. Since the similarity of the data objects in the same class is high, the class boundary is selected. The data segment is used to build a check cache and also provides a good deduplication ratio [27].

According to the problem description, the redundant data are filtered out by using the check metadata deduplication algorithm. Firstly, the weighted dataset is clustered, and then the compressed neighboring algorithm is used to obtain the weighted subset, and the similarity metadata are eliminated based on the weighted subset, thereby reducing the size of the index. Eliminating metadata with high similarity can effectively reduce the amount of metadata and further reduce system resource overhead while maintaining the deduplication ratio [28, 29].

The whole process of deduplication is divided into two parts: data segment similar clustering and deduplication.

In the similar clustering phase, for the subsequent description to be clear, the subsequent symbols are uniformly defined: the *sim* fingerprint set defining the data to be checked is $S' = \{s'_1, \dots, s'_n\}$, the cluster before the similar clustering is defined as $C' = \{C'_1, \dots, C'_K\}$, the cluster after clustering is $C'' = \{C''_1, \dots, C''_K\}$, a distance measure between two similar data segments is defined as $\text{dist}(s'_i, s'_j)$, and the

FIGURE 2: Description of data redundancy problem.

Hamming distance between two *sim* fingerprint values is $\text{Ham}(s'_i, s'_j)$.

Here, the saved *sim* fingerprint value information is used to represent a data object, and the fingerprint set $S' = \{s'_1, \dots, s'_n\}$ of the check data is obtained. Then, the data objects in the set $S' = \{s'_1, \dots, s'_n\}$ are clustered, and the data segments are divided into K -class $C'' = \{C''_1, \dots, C''_K\}$.

The distance measure between two similar data segments is expressed as the Hamming distance of the *sim* fingerprint values, which is

$$\text{dist}(s'_i, s'_j) = \text{Ham}(s'_i, s'_j). \quad (10)$$

Therefore, the entire clustering process can be described as follows:

- (1) Select K representative objects $\{b_1, \dots, b_K\}$ from S' as the initial center point.
- (2) Assign each remaining object to the cluster $C' = \{C'_1, \dots, C'_K\}$ represented by its nearest central point.
- (3) For each cluster, the l noncentral point objects s'_j in the cluster are traversed. The total cost of replacing center point b_i with s'_j is calculated:

$$\cos t = \sum_{j=1}^l \text{dist}(b_i, s'_j) = \sum_{j=1}^l \text{Ham}(b_i, s'_j). \quad (11)$$

The object with the lowest total cost is selected as the new center point.

- (4) Repeat steps (2) and (3) until the K center points no longer change. The K cluster $C'' = \{C''_1, \dots, C''_K\}$ is the obtained, that is, the required K class similar data.

In the deduplication phase, the check weight subset is defined as $S'' = \{s''_1, \dots, s''_n\}$, and the process of the deduplication phase is as follows.

The *sim* fingerprint value of the data segment is used to replace the data segments, and the *sim* fingerprint value set $S' = \{s'_1, \dots, s'_n\}$ of all data segments is obtained. Two memories st and gr for the set $S' = \{s'_1, \dots, s'_n\}$ are set. All the

samples of $S' = \{s'_1, \dots, s'_n\}$ are put into gr , and the fingerprint value of a data segment *sim* is randomly extracted from gr and put into st . A fingerprint value s'_K of a data segment *sim* is randomly extracted from gr , and the *sim* fingerprint value in st is used as the reference set. s'_K is classified, and a closest s'_{nK} to s'_K from st is found. Assuming $s'_{nK} \in C'_i$ is the same as $s'_K \in C'_i$, then it thinks that the redundancy judgment is correct, and to delete s'_K ; otherwise, s'_K is stored as a new category in st . The above steps for all samples in gr are performed until gr is empty. The set after removing redundant data at this time is $S''' = \{s'''_1, \dots, s'''_n\}$.

According to the above process, redundant data in the real-time streaming big data can be cleared.

2.3. Controllable Clustering for Associated Real-Time Streaming Big Data Based on Multi-Source Data Fusion. Assume that the fusion range of multi-source data fusion is $[q_{\min}, q_{\max}]$, where $q_{\min}$ represents the minimum number of neighbors and $q_{\max}$ represents the maximum number of neighbors.

The q nearest neighbors of N'' search for n'' data points are mapped to a similarity function *similarity*, and the attribute similarity of the data point (z_1, z_2) can be recorded as similarity (z_1, z_2) , namely:

$$\text{similarity}(z_1, z_2) = \frac{n'' z_1 z_2}{2N''}. \quad (12)$$

The similarity function of equation (12) is updated: for data point (z_1, z_2) , assume z_1 is in q neighbors of z_2 , or vice versa:

$$\text{similarity}(z_1, z_2) = \text{similarity}(z_1, z_2) + \frac{n'' z_1 z_2}{2N''}. \quad (13)$$

The loop variable $t'' = t'' + 1$; assuming $t'' < N''$, the similarity function is reconstructed; otherwise, the following steps are performed.

According to the similarity function formed by the above process, the multi-source data fusion concept is used to construct the real-time streaming big data fusion device, and the output result is the final clustering result:

$$P^* = S''' \otimes \text{similarity}(q_i, q_j) \cdot \delta, \quad (14)$$

where $\otimes$ represents the connector of attribute similarity data in the real-time streaming big data, δ represents the clustering factor, and the value controlled in $[0.6, 0.7]$ can improve the clustering precision.

3. Results

In order to verify the effectiveness of the controllable clustering algorithm for associated real-time streaming big data based on multi-source data fusion, an experiment is conducted. The experimental hardware environment is shown in Figure 3.

The algorithm is implemented by using C programming under Linux and the associated real-time streaming database. The experimental indicators are as follows:

FIGURE 3: Experimental environment.

(a)

(b)

FIGURE 4: Continued.

FIGURE 4: Comparison of clustering security of different algorithms. (a) Big data clustering security based on local key nodes. (b) SOM hybrid attribute data clustering security based on heterogeneous value difference metric. (c) Controllable clustering security of associated real-time streaming big data based on multi-source data fusion.

FIGURE 5: Comparison of data clustering accuracy of different algorithms. (a) SOM hybrid attribute data clustering accuracy based on heterogeneous value difference metric. (b) Data classification accuracy based on Spark-FCM. (c) Improved manifold clustering accuracy based on density peak search. (d) Controllable clustering accuracy of associated real-time streaming big data based on multi-source data fusion.

FIGURE 6: Comparison of data clustering time consumption of different algorithms. (a) Time consumption of data classification based on Spark-FCM. (b) Time consumption of improved manifold clustering based on density peak search. (c) Time consumption of big data clustering based on local key nodes. (d) Time consumption of controllable clustering for associated real-time streaming big data based on multi-source data fusion.

- (1) Process security of data clustering.
- (2) Clustering accuracy.
- (3) Time consumption of clustering.
- (4) Energy consumption of data clustering.

The experimental results are as follows.

Analysis of Figure 4 shows that the big data clustering algorithm based on local key nodes and the SOM hybrid attribute data clustering algorithm based on heterogeneous value difference metrics have lower operational safety factors. The controllable clustering algorithm for associated real-time streaming big data based on multi-source data fusion has higher data clustering security coefficient under different types of attack data. This is mainly because the algorithm uses the particle swarm optimization algorithm to detect the anomaly data before data clustering and eliminates it, which effectively enhances the security performance of the algorithm.

Analysis of Figure 5 shows that the proposed algorithm is superior to the current data clustering algorithm in data clustering accuracy. The algorithm uses filtering technology to suppress network interference and initially improves data

clustering accuracy. The clustering factor is introduced to further improve the data clustering accuracy of the proposed algorithm.

In Figures 6 and 7, the clustering time of using the controllable clustering algorithm for real-time streaming big data based on association of multi-source data fusion is not affected by the clustering data amount, the clustering efficiency is high, and the overall energy consumption of clustering is lower. This is mainly because the algorithm reduces the time spent on data clustering, the energy consumption of clustering, and the burden of data clustering by eliminating redundant data, effectively controlling the time and energy consumption of data clustering and enhancing the overall performance of the proposed algorithm.

4. Discussion

In the discussion, the clustering factor δ is taken as the discussion object, and the influence of its value range on data clustering is observed. Matlab2017a is used to simulate the influence of the value change on the data clustering accuracy rate. It is observed that the value is controlled within the

FIGURE 7: Comparison of data clustering energy consumption of different algorithms. (a) Improved manifold clustering energy consumption based on density peak search. (b) Energy consumption of SOM hybrid attribute data clustering based on heterogeneous value difference metric. (c) Energy consumption of data classification based on Spark-FCM. (d) Energy consumption of controllable clustering for associated real-time streaming big data based on multi-source data fusion.

FIGURE 8: Effect of δ range of values on data clustering accuracy.

interval of $[0.6, 0.7]$, and the data clustering accuracy rate is the highest, that is, δ can effectively improve the clustering accuracy in this interval. The results are as follows.

It can be seen from Figure 8 that when the value of δ is $[0.4, 0.6]$, the data clustering accuracy rate is generally improved, but it is not very ideal; when the value of δ is $[0.7, 0.9]$, the data clustering accuracy rate is decreasing;

when the value of δ is at $[0.6, 0.7]$, the data clustering accuracy rate is the highest, about 98%. It can be observed from the data comparison that the value of δ has a great influence on the accuracy of data clustering, and when the value of δ is $[0.6, 0.7]$, it can control the clustering accuracy to the best, indicating that the value range is reliable.

5. Conclusions

Efficient clustering of large-scale data is very important for data utilization. At present, the performance of relevant research results is to be improved, and a controllable clustering algorithm for associated real-time streaming big data based on multi-source data fusion is proposed. By detecting and eliminating abnormal data and redundant data, it lays a foundation for data clustering and realizes data clustering by constructing a correlated real-time streaming big data fusion device. The experimental results show that the proposed algorithm has strong clustering performance and is feasible.

In the next step, the following aspects can be regarded as the research focus: data clustering technology is constantly innovating, including data clustering system, which can

integrate clustering system with clustering algorithm and use clustering algorithm to control clustering system to further improve data clustering performance.

Data Availability

The datasets used and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] J.-Y. Chen and H.-H. He, "A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data," *Information Sciences*, vol. 345, no. C, pp. 271–293, 2016.
- [2] N. Taherkhani and S. Pierre, "Centralized and localized data congestion control strategy for vehicular Ad Hoc networks using a machine learning clustering algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3275–3285, 2016.
- [3] H.-Y. Hsieh, T.-C. Juan, Y.-D. Tsai, and H.-C. Huang, "Minimizing radio resource usage for machine-to-machine communications through data-centric clustering," *IEEE Transactions on Mobile Computing*, vol. 15, no. 12, pp. 3072–3086, 2016.
- [4] A. Rodriguez, A. Tosiya, B. Kim et al., "Patent clustering and outlier ranking methodologies for attributed patent citation networks for technology opportunity discovery," *IEEE Transactions on Engineering Management*, vol. 63, no. 4, pp. 426–437, 2016.
- [5] W. Gao, M. R. Farahani, A. Aslam, and S. Hosamani, "Distance learning techniques for ontology similarity measuring and ontology mapping," *Cluster Computing-the Journal of Networks Software Tools and Applications*, vol. 20, no. 2SI, pp. 959–968, 2017.
- [6] Z. Li, C. Han, and T. Gu, "Economics of biomass gasification: a review of the current status," *Energy Sources, Part B: Economics, Planning and Policy*, vol. 13, no. 2, pp. 137–140, 2018.
- [7] X. Wang and N. Wan, "Cloud pattern clustering algorithms on big data of internet of things based on extracting chaotic correlation features of events," *Application Research of Computers*, vol. 38, no. 2, pp. 391–397, 2021.
- [8] E. Bellini, G. Morgari, and M. Coppola, "An extension of cook's elasti cipher," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 20, no. 5, pp. 1151–1171, 2017.
- [9] K. Subklay and N. Pochai, "Numerical simulations of a water quality model in a flooding stream due to dam-break problem using implicit and explicit methods," *Journal of Interdisciplinary Mathematics*, vol. 20, no. 2, pp. 461–495, 2017.
- [10] Y. Cao and X. D. Qian, "A big data clustering algorithm based on local key nodes," *Computer Engineering and Science*, vol. 38, no. 7, pp. 1338–1343, 2016.
- [11] Y. Jiang and X. Li, "Broadband cancellation method in an adaptive co-site interference cancellation system," *International Journal of Electronics*, 2021.
- [12] X. Wu, W. Zheng, X. Xia, and D. Lo, "Data quality matters: a case study on data label correctness for security bug report prediction," *IEEE Transactions on Software Engineering*, vol. 1, 2021.
- [13] W. Zheng, Y. Xun, X. Wu, Z. Deng, X. Chen, and Y. Sui, "A comparative study of class rebalancing methods for security bug report classification," *IEEE Transactions on Reliability*, vol. 70, no. 4, pp. 1–13, 2021.
- [14] Y. X. Zhang, H. D. Peng, and J. H. Wang, "Self-organizing mapping clustering algorithm based on heterogeneous value difference metric for mixed attribute data," *Chinese Journal of Scientific Instrument*, vol. 37, no. 11, pp. 2555–2562, 2016.
- [15] G. L. Wang, G. L. Zhou, C. R. L. Sa, and Y. L. Zhu, "Parallel fuzzy C-means clustering algorithm in spark," *Journal of Computer Applications*, vol. 36, no. 2, pp. 342–347, 2016.
- [16] Z. Lv, Y. Li, H. Feng, and H. Lv, "Deep learning for security in digital twins of cooperative intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2021.
- [17] Z. Lv, D. Chen, H. Feng, H. Zhu, and H. Lv, "Digital twins in unmanned aerial vehicles for rapid medical resource delivery in epidemics," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–9, 2021.
- [18] Z. Lv, L. Qiao, and A. K. Singh, "Fine-grained visual computing based on deep learning," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 17, no. 1, 2021.
- [19] Y. L. Liu and J. P. Zhang, "Improved manifold clustering algorithm based on density peaks search," *Computer Engineering and Design*, vol. 37, no. 6, pp. 1654–1658, 2016.
- [20] Y. Q. Sun, Y. Y. Li, and Y. Lu, "Optimization of K-means clustering algorithm based on MapReduce," *Computer Measurement & Control*, vol. 24, no. 7, pp. 272–275, 2016.
- [21] Z. Lv, D. Chen, and Q. Wang, "Diversified technologies in internet of vehicles under intelligent edge computing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2048–2059, 2021.
- [22] Z. Lv, L. Qiao, M. S. Hossain, and B. J. Choi, "Analysis of using blockchain to protect the privacy of drone big data," *IEEE network*, vol. 35, no. 1, pp. 44–49, 2021.
- [23] Z. Lv, D. Chen, R. Lou, and H. Song, "Industrial security solution for virtual reality," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6273–6281, 2021.
- [24] D. Q. Wang and X. X. Wang, "Large data optimization particle swarm clustering algorithm based on cloud storage," *Electronic Design Engineering*, vol. 25, no. 2, pp. 26–30, 2017.
- [25] F. Deng, "Multiple hops network classification attribute data fuzzy clustering in the simulation," *Computer Simulation*, vol. 34, no. 1, pp. 292–295, 2017.
- [26] J. M. Dong, M. Pan, and C. Zhang, "Parallel subspace clustering using MapReduce," *Science Technology and Engineering*, vol. 17, no. 15, pp. 104–110, 2017.
- [27] H. Long and X. M. Zhang, "Large data clustering algorithm based on modified two order cone programming model," *Bulletin of Science and Technology*, vol. 32, no. 8, pp. 168–171, 2016.
- [28] X. L. Zhou and H. Zhang, "The study on the model of periodic BIT data analysis relation to time sequence," *Journal of China Academy of Electronics and Information Technology*, vol. 12, no. 2, pp. 128–131, 2017.
- [29] Z. Ma, J. Xie, H. Li et al., "The role of data analysis in the development of intelligent energy networks," *IEEE Network*, vol. 31, no. 5, pp. 88–95, 2017.