Research Article

An Improved Algorithm Based on Fast Search and Find of Density Peak Clustering for High-Dimensional Data

Hui Du , Yiyang Ni , and Zhihe Wang

The School of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

Correspondence should be addressed to Hui Du; duhuiywy@nwnu.edu.cn

Received 1 April 2021; Revised 1 June 2021; Accepted 6 July 2021; Published 27 July 2021

Academic Editor: Xingsi Xue

Copyright © 2021 Hui Du et al. 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.

The find of density peak clustering algorithm (FDP) has poor performance on high-dimensional data. This problem occurs because the clustering algorithm ignores the feature selection. All features are evaluated and calculated under the same weight, without distinguishing. This will lead to the final clustering effect which cannot achieve the expected. Aiming at this problem, we propose a new method to solve it. We calculate the importance value of all features of high-dimensional data and calculate the mean value by constructing random forest. The features whose importance value is less than 10% of the mean value are removed. At this time, we extract the important features to form a new dataset. At this time, improved t-SNE is used for dimension reduction, and better performance will be obtained. This method uses t-SNE that is improved by the idea of random forest to reduce the dimension of the original data and combines with improved FDP to compose the new clustering method. Through experiments, we find that the evaluation index NMI of the improved algorithm proposed in this paper is 23% higher than that of the original FDP algorithm, and 9.1% higher than that of other clustering algorithms (*K*-means, DBSCAN, and spectral clustering). It has good performance in high-dimensional datasets that are verified by experiments on UCI datasets and wireless sensor networks.

1. Introduction

In our daily life, when we are faced with a problem that we do not have an accurate standard to classify. Cluster analysis is an effective means to judge and analyze. It has applications in many fields, such as finance, medical, and image. The cluster algorithm divides elements into clusters according to calculated mathematical characteristics. Although many clustering algorithms have been studied and discussed, there is no agreement on the definition of clustering. Speaking of clustering algorithm, the first thing we have to mention is *K*-means [1]. *K*-means is an efficient and concise algorithm, which has been discussed by many researchers. It only needs to set the number of clusters to calculate clustering results to users. But these methods cannot detect clusters for nonspherical data [2]. DBSCAN [3] is also a classic and effective algorithm. It is a representative clustering algorithm based on density. DBSCAN is an algorithm that is in line with the spatial distribution of data and the consistency of data density. The author creatively defines two parameters to constrain

and control the generation of clusters. But these two parameters also lead to the effect of the DBSCAN algorithm seriously affected by the parameters. It has good robustness [4] to outliers and can even detect outliers. Spectral clustering [5] is a method of clustering data points by means of discrete mathematics. The algorithm constructs an undirected graph and judges and analyzes the properties of the undirected graph. The most ingenious and important part of this algorithm is to construct Laplacian matrix. To construct Laplacian matrix, Laplacian matrix needs to calculate the similarity matrix. Firstly, the similarity matrix will use full connection mode. There are many kernel functions to measure the relationship between points. The average effect of Gaussian kernel function is the best among many kernel functions. Secondly, the Laplacian matrix is constructed by calculating adjacency matrix and degree matrix through similarity matrix. Finally, it needs to get the eigenvector of Laplacian. According to eigenvalues and eigenvectors, we can use other clustering algorithms to complete the clustering task. Balanced iterative reducing and clustering using

hierarchies (BIRCH) [6] is method by tree structure to cluster quickly. Birch algorithm is to form a clustering feature (CF) tree. By calculating the similarity of the dataset, the most similar sample data points in the dataset are combined, and the process is iterated [7]. Fuzzy clustering [8] is a new clustering algorithm with the development of the automation control field. In the theory of fuzzy mathematics, the concept of membership degree is mentioned for the first time. This theory greatly promotes its development. The main solution of the algorithm is to introduce membership degree to optimize or even solve problem when a point cannot be effectively identified. Affinity propagation (AP) is a clustering algorithm using a voting mechanism with good results. The idea of AP algorithm is very interesting, which is illustrated by examples in our life. For example, there are many commodities and many customers. If a commodity can be selected by customers, it must have its own strength. That is to say, it must have enough attraction. Secondly, it is the direct word-of-mouth of customers. One customer says yes, and further recommends the next customer to buy it, and so on. In this case, the commodity is like the cluster center to be selected, and the customer is the point attached to the cluster center. To sum up, the core of AP clustering algorithm is to calculate the attribution matrix and attraction matrix.

After introducing FDP, there are many papers that solve FDP's problem. This paper proposes a method [9] that uses the characteristics of data point density distribution to cluster efficiently and quickly. The algorithm ingeniously designs a set of calculation method and selects the point with the highest density in a certain area as the center. Noncentral point assignment is based on the class of the nearest point whose density is larger than itself. Outliers are found and excluded from the analysis. Regardless of the shape of the cluster and the dimension of the embedded space, clustering will be recognized.

However, FDP cannot solve many data features or high-dimensional data effectively, some nonmain features will interfere and affect the performance of the algorithm, and the scientists have made a series of improvements. Among them, dimension reduction is the first choice. A new fast hybrid dimension reduction method [10] is proposed. It innovatively proposes a method that combines multiple feature extraction methods. In this way, the interference of nonmain features can be reduced and the efficiency can be improved. The Fisher score and feature selection based on information gain are used to remove nonmain features. In this way, the interference of nonmain features is reduced, and the clustering effect is significantly improved. At present, there are many ways to reduce dimension, and principal component analysis (PCA) [11] is the best one. PCA creatively maps high-dimensional features to the low-dimensional features, so that the new low-dimensional data is an orthogonal matrix, which is also called the main component. Because it is a low-dimensional feature reconstructed on the basis of the original high-dimensional feature, the original feature is kept while the dimension is reduced. In conclusion, we compare with the PCA in experiments.

The purpose of studying the clustering algorithm is to serve people's life. In this paper [12], the popular concept of topology is used. In topology, manifold learning is the key to this paper. Manifold learning is currently a popular dimension reduction method. Let us briefly introduce manifold learning. For example, a piece of paper can be seen as two-dimensional when it is tiled. But when it is kneaded into a mass, it can be regarded as three-dimensional. For any point on the tissue, whether it is kneaded into a ball or flat, its relative position has not changed. If we can reduce the dimension smoothly and keep the feature unchanged. At the same time, the index matrix of the discrete clustering optimization process is easily affected by noise. To solve matters, a new clustering way was proposed in this paper. It combined local adaptive subspace learning and knowledge clustering to mine discriminant information adaptively.

Aiming at the practical problems of sample data nonlinearity and high dimension in complex system evaluation or prediction, in paper [13], they all used a memetic algorithm model [14] to achieve dimensionality reduction and achieved good results.

In another paper, it proposed a clustering algorithm for high-dimensional stream data. This algorithm introduced dimension reduction into the framework of stream clustering. When the new data arrives, for the sake of finding the local shadow space, there is necessary processing of the disordered new data. It is very necessary to reduce the high dimension to the low dimension. The algorithm innovatively used the unsupervised linear discriminant analysis (LDA) to process the data, so as to find the local shadow space. The obtained local subspace will maximally separate the adjacent microclusters from the incident point. Introduction points are admeasured to the microclusters in the projection space, which can be improved by this method.

Facing high-dimensional data, we proposed a method to combine FDP to achieve the performance of FDP on high-dimension data.

At this moment, let us discuss the practical application of clustering. When it comes to practical application, we have to mention the Internet of Things (IoT). It is often used in the intelligent world. As a sensing layer, wireless sensor networks are composed of many sensor networks. The sensor is the most important part of Internet of Things. Sensors are like human facial features to perceive the world. Sensors constantly provide information for users to facilitate their production and life. Therefore, to obtain a stable network, the energy control of the sensor must be optimized first, so as to extend the life of each sensor in the network. Sensor in the network is like a data point in space. At this time, each data point has a series of characteristics, such as relative distance and energy consumption. How to better control the energy consumption of wireless sensor is the difficult problem.

In this paper, [15] creatively introduced the semantic relationship between sensors to promote the effect of the whole network. Therefore, it proposed the new sensor ontology integration technology by introducing such a mechanism which is called the debate mechanism (DM). The purpose of this method is to extract sensor ontology alignment [16, 17].

In this way, the communication ability between wireless sensors is strengthened, and the performance of the whole network is improved. The global factor in the algorithm is calculated by the correctness factor. The local factor is calculated through the debate mechanism. By getting the global factor and the local factor, the judgment factor can be obtained by combining them. The judgment factor is used to achieve ontology alignment. By this means, the effect of the whole wireless sensor network can be improved. Inspired by this article, we finally apply the improved FDP to wireless sensor networks and do a comparative experiment to show the performance.

2. Related Work

2.1. Introduction to FDP. ρ_i and d_{ij} are important values of FDP. Equations (1) and (2) are designed to evaluate local density.

$$\rho_i = \sum_j \chi(d_{ij} - d_c), \quad (1)$$

$$\chi(x) = \begin{cases} 1, & x < 0, \\ 0, & x = 0, \end{cases} \quad (2)$$

where ρ_i is used to represent the local density. The degree of similarity between point i and point j is expressed by d_{ij} . It is generally measured by Euclidean distance. For Equation (1), d_c is usually set to 1%-2% of the total number of samples (the total number of data points). Although the original text does not specifically point out, the setting of d_c needs to be set by the user, and the impact on the algorithm is very huge. Partly, the setting of d_c is also very difficult. This parameter has a great influence on the algorithm. Moreover, if the setting is not correct, the expected effect will not be obtained, and it is mistaken for the performance of the algorithm itself. Its value setting is also a very popular direction, and its value setting can effectively improve the performance of FDP.

Secondly, δ_i is the nearest Euclidean distance at all points with a greater density than itself. The decision graph is set by ρ_i and δ_i as the x -axis and y -axis of the coordinate axis, respectively. The larger and larger sample points are selected as the center of the class cluster in the decision graph. It needs to check manually the region according to the generated decision graph, and the point in the selected region is the cluster center point. The cluster of each noncenter sample point is the cluster of the nearest sample point higher than the point in the neighborhood.

2.2. Introduction to t -SNE. Among many dimensionality reduction algorithms for high-dimensional data, the stochastic neighbor embedding (SNE) [18] is a very special one. The idea of its algorithm is very clever and simple, but it contains a lot of probability theory. It is very similar for the popular learning mentioned above. It is about flattening a spatial dimension into a plane. The algorithm first describes the distribution of each point; how to measure the distribution is a key. There are many ways to measure; here, we choose

Euclidean distance first. According to the research, different measurement methods have a great impact on the performance of the algorithm. We consider two data points x_i and x_j which are high-dimensional data. The conditional probability represents the degree to which data x_j is a neighbor of data x_i by p_{ji} . We can define p_{ji} in this way.

$$p_{ji} = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}, \quad (3)$$

where σ_i is a statistical constant. First, we get the Gaussian distribution x_i which is centered and then calculate its variance.

The essence of dimension reduction is not to change the nature of data points and the relationship between data points. Low-dimensional data can keep as many features as possible. y_i and y_j are the data in low-dimensional corresponding to high-dimensional x_i and x_j separately. Through such a mapping relationship is to establish the dimensionality reduction process. The conditional probability density of a low-dimensional is defined just like that of a high-dimensional by q_{ji} .

$$q_{ji} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq i} \exp(-\|y_i - y_k\|^2)}. \quad (4)$$

It is obvious that two conditional probability densities are obtained by calculation, and how to deal with and use these two values will be the key of this algorithm. If the high- and low-dimensional distributions are consistent, then the two conditional probabilities will be equal. Then, our goal is relatively clear. When the two conditional probabilities are equal or the difference is very small, the effect of dimension reduction will be perfect. The author of SNE introduced the Kullback-Leibler (KL) divergence distance to solve this problem.

$$\sum_i KL(P_i \| Q_i) = \sum_i \sum_j p_{ji} \log \frac{p_{ji}}{q_{ji}}. \quad (5)$$

The dimension reduction effect of SNE is catastrophic for clustering. The clustering algorithm cannot cluster the reduced data effectively. Such dimension reduction is meaningless for clustering. Because SNE pays more attention to the local structure and ignores global structure. Having a certain impact, there is also congestion problem when using symmetric SNE.

The performance of t -distribution and Gaussian distribution is similar without interference data. But when there are outliers in the sample, there are inconsistencies. The simulation result of Gaussian distribution is not as good as that of t -distribution. t -distribution can keep the internal structure of the original data unchanged, and the variance is small. t -distribution can keep and show the characteristics of the original data better. At this time, the author can

FIGURE 1: t-SNE for 0-5 digits. We select digits dataset 0-5 and use t-SNE to test the effect. When there are few features, it can be seen from the figure that only 2 small clusters are not distributed according to the ideal.

FIGURE 2: t-SNE for 0-6 digits. When the features increase, the data reduced by t-SNE is more difficult to be analyzed by clustering.

solve this problem by using t -distribution instead of Gaussian distribution. Equation (6) shows the result after replacing with t -distribution.

$$q_{ji} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq i} \left(1 + \|y_i - y_k\|^2\right)^{-1}}. \quad (6)$$

KL distance is to find the optimal value.

$$\frac{\delta C}{\delta y_i} = 4 \sum_j \left(p_{ij} - q_{ij}\right) (y_i - y_j) \left(1 + \|y_i - y_j\|^2\right)^{-1}. \quad (7)$$

However, we find that the FDP algorithm cannot effectively cluster the data after t-SNE dimensionality reduction.

In the process of Figures 1–3, the data in Figure 1 only has six groups of features. After dimensionality reduction, we can see that each class is separated and not mixed together. Figure 2 has 7 groups of features. At this time, a

small green piece appears near a large yellow block, and a small yellow one appears near the green one. If a clustering algorithm is used, the green class will be divided into yellow, and the yellow small piece is divided into green, which leads to the final result error of clustering. In this picture, the same is true. There are many other cases which will lead to the error of the final clustering results. But there are 10 groups of data in Figure 3. It should have appeared in the same region with the same color. However, it can be seen from the graph that many colors are mixed together and are not effectively separated. If clustering algorithm is used at this time, the accuracy of clustering algorithm will be reduced, and even the wrong results will be obtained. In the next section, we propose an improvement.

3. Algorithm Improvement

3.1. Feature Extraction. Figures 1–3 show the process in which the effect of the algorithm becomes worse as the feature increases. t-SNE [19] has decreased with the increase of data features. The cluster algorithm cannot be correctly

FIGURE 3: t-SNE for 0-9 digits. When the features are further increased, the distribution of data is more unclear, which is not conducive to clustering algorithm.

distributed to the right location. This will inevitably degrade the performance of cluster algorithm. We decided to introduce random forest to improve the performance.

The performance of traditional t-SNE still has some problems, so we decided to introduce random forest to improve the performance. The data in scientific research and production life are very complex, but the main features of these data that can distinguish each other are sometimes not many. For example, fruit can be distinguished by shape and color. Therefore, effectively selecting the main features can improve the performance of the t-SNE algorithm. The feature extraction of random forest [20] is mainly based on the out of bag (OOB) principle. If a feature is important, then when a certain amount of noise is introduced into the distributed data of this feature, the performance of the model should be greatly changed by random forest training with only the changed data of this feature. On the contrary, if a feature is unimportant, the performance of the retrained model will not change much. Data has many features. We need to calculate the importance value of all features. Firstly, a random forest is established, and the decision tree of the random forest uses C4.5. Firstly, for a feature, the error of its packet data is calculated according to the established random forest, which is recorded as err_{OOB1} . Then, the interference data is added to the same feature to calculate err_{OOB2} .

$$Importance_x = \frac{\sum_{i=1}^N (err_{OOB2} - err_{OOB1})}{N}. \quad (8)$$

N stands for n trees. If we add noise to a feature at random, $Importance_x$ will be greatly reduced, which means that the main feature. It has a high degree of importance. The feature select process is to sort the feature variables by random forest in descending order of $Importance_x$. A new feature set is obtained by determining the deletion ratio and eliminating the unimportant indexes from the current feature variables. After calculating the importance value of all the features, we calculate their average value. If the importance

```

1: Input:  $D = \{x_1, x_2, \dots, x_m\}^n$ .
2: Output:  $Y = \{y_1, y_2, \dots, y_m\}^2$ .
3: Build random forest.
4: Compute  $Importance_x$ .
5: Compute  $avg(Importance_x)$ .
6: for 1 to  $n$ .
7:   if  $Importance_i \leq avg$ .
8:     Remove feature.
9:   end.
10: end.
11: Generate  $G = \{x_1, x_2, \dots, x_m\}^r$ .
12:  $G$  use by t-SNE.
13: Generate  $Y = \{y_1, y_2, \dots, y_m\}^2$ .

```

ALGORITHM 1: The improvement of t-SNE.

value is less than 10%-20% of the mean value, the remaining features are the main features. After dimensionality reduction of these features, the effect of FDP will increase. Figures 4 and 5 show the effect after extracting the main features. t-SNE will further put the original class together instead of leaving a class. t-SNE PCA and locally linear embedding [21] (LLE) cannot effectively separate the data, which affects the clustering effect.

Now, let us take a look at Figure 4. This picture shows the comparison between the improved t-SNE and t-SNE and PCA and LLE. In Figure 4, the improved t-SNE, t-SNE, PCA, and LLE are represented by (a), (b), (c), and (d), respectively. We use the wine dataset in the UCI dataset, which has 3 categories and 13 features. The goal of this experiment is to reduce this set of data to 2 dimensions. We use the same color to represent the same class and use four algorithms to reduce the original data to 2 dimensions. It can be seen that t-SNE, PCA, and LLE do not effectively divide the same color into the same area. In t-SNE, in the -200 to 200 regions, three colors are mixed together. In PCA, in the -200 to 200 regions, the three colors are mixed together, so is LLE. In the improved t-SNE, green, purple, and blue are effectively

(a)

(b)

(c)

FIGURE 4: Continued.

FIGURE 4: Comparison chart in wine datasets. (a) Improved t-SNE. (b) t-SNE. (c) PCA. (d) LLE.

divided into three parts. In this way, the clustering algorithm can cluster more effectively.

In Figure 5, the improved t-SNE, t-SNE, PCA, and LLE are represented by (a), (b), (c) and (d), respectively. We use the digits dataset in the UCI dataset, which is 64-dimensional data. The goal of this experiment is to reduce this set of data to 2 dimensions. We use the same color to represent the same class and use four algorithms to reduce the original data to 2 dimensions. First of all, you can see that all the colors in c are mixed up in a disorderly way. After dimensionality reduction, each class is not separated effectively. This will lead to the subsequent clustering effect worse or even get the wrong result. In (d), although each color is roughly in a straight line, except for fans and dark blue which are obviously separated, other colors are mixed together and not effectively separated. The effect in (b) is better than that in (c) and (d), but we can see that a large number of small pieces are not assigned to the corresponding position. For example, a small knob of light blue is far away from the original a lumpen mass of light blue, and this small knob of light blue is closer to other colors, which will lead to the subsequent clustering algorithm cannot effectively classify. It will lead to the result error. The improved algorithm can reduce the occurrence of two cases, so it can improve the accuracy of clustering algorithm after dimension reduction.

3.2. Self-Adaptive Selection. Now, let us talk about FDP. In our further experiments, we found a lot of problems and defects about the algorithm itself. The selection of d_c and the judgment of the decision graph will affect the result of the final clustering algorithm. In this paper, we introduce and improve the decision graph selection problem. We will further improve the selection of d_c in the follow-up work. The improvement of d_c selection is also very meaningful. Back to the part of the decision graph, we find that the experimenter needs to decide the cluster center by himself. This greatly affects the clustering performance and effect. When artificial selection is introduced into the algorithm, the accuracy of the algorithm will be greatly reduced. If this algorithm needs to be promoted and expanded, its ease of use will be

greatly reduced. Through a large number of experiments, we solve the problem of manually selecting the center point and improve the accuracy of the original algorithm.

For the problem that the center point cannot be selected effectively in FDP, we design a new judgment system through the difference and change rate, so that it can select the center adaptively. It does not need to check manually to complete the center point selection. We should make full use of ρ_i and δ_i . Firstly, ρ_i and δ_i are multiplied, and the value is set as λ_i

$$\lambda_i = \rho_i \cdot \delta_i, \quad (9)$$

$$\Delta_i = \lambda_i - \lambda_{i+1}, \quad (10)$$

$$\theta_i = \frac{\max(\Delta_i, \Delta_{i+1})}{\min(\Delta_i, \Delta_{i+1})}. \quad (11)$$

The improved algorithm relies on the calculation of the λ . We sort and start with 1. λ_1 is biggest, and so on. And then, we make use of mathematical analysis of the results λ . Firstly, we use Equation (10) to get the difference. The main idea of formula (10) is to make difference Δ between two adjacent terms. Then, the calculated difference Δ is processed by Equation (11). The main idea of Equation (11) is to get the change rate of adjacent difference θ . These two values are regarded as the prerequisite of the core point judgment.

This is the pseudocode described by the algorithm RCD-FDP.

At this time, we calculate the arithmetic mean value of Δ and θ to get $\text{avg}(\Delta)$ and $\text{avg}(\theta)$, respectively. At this point, we can transform the problem into comparing Δ and θ with $\text{avg}(\Delta)$ and $\text{avg}(\theta)$. After a series of experiments, when Δ is less than 10%-25% of the $\text{avg}(\Delta)$ and θ is less than 50% of the $\text{avg}(\theta)$. You can stop judging and get the center point. Table 1 is a manual dataset. Due to the large amount of data, only the first 11 sample data are shown in this paper. It can be seen that if we judge by hand, we cannot accurately judge the cluster center. However, by introducing Δ , θ , $\text{avg}(\Delta)$, and $\text{avg}(\theta)$, the core point can be determined adaptively. It avoids

(a)

(b)

(c)

FIGURE 5: Continued.

FIGURE 5: Comparison chart in digits datasets. (a) Improved t-SNE. (b) t-SNE. (c) PCA. (d) LLE.

```

1: Input:  $D = \{x_1, x_2, \dots, x_m\}^n$ ,  $\Delta$ ,  $\theta$ .
2: Output: cluster class  $y_i$  belong to  $x_i \in D$ .
3: Compute and sort  $\lambda_i$ .
4: for 1 to  $m$ .
5:    $\Delta_i = \lambda_i - \lambda_{i+1}$ .
6: end.
7: for 1 to  $m$ .
8:    $a = \max(\Delta_i, \Delta_{i+1})$ .
9:    $b = \min(\Delta_i, \Delta_{i+1})$ .
10:   $\theta_i = a/b$ .
11: end.
12: for 1 to  $m$ .
13:   if  $\Delta_i < \Delta$  &&  $\theta_i < \theta$ .
14:     cluster number =  $i - 1$ .
15:   end.
16: Cluster by FDP.

```

ALGORITHM 2: RCD-FDP.

the ground error caused by manual judgment. The above pseudocode RCD-FDP shows the process of the algorithm.

Combining the data after improved t-SNE dimension reduction with RCD-FDP, this algorithm is called IT-RCD-FDP, which has an obvious effect on high-dimensional datasets.

4. Experiment

4.1. Evaluation Criterion. This section shows the performance of the improved algorithm through experimental comparison. In this experiment, digits, wine, and heart disease in the UCI dataset are selected. Through these datasets, the experiment uses some commonly used evaluation criteria [22] to compare with the original FDP algorithm, K -means algorithm, DBSCAN, and spectral clustering. Four evaluation criteria are used in this experiment. It includes accuracy, adjusting rand index, normalized mutual information, and completeness.

TABLE 1: Change rate and difference.

i	ρ	δ	Δ	θ
1	17.1	21.15	240.455	12.4
2	10.91	11.11	19.38	4
3	8.53	11.95	79.055	8.7
4	6.1	3.75	8.995	1.3
5	4.13	3.36	6.46	3.8
6	5.26	1.41	1.668	8.9
7	7.19	0.8	0.186	2.1
8	5.06	1.1	0.406	2.3
9	1.66	3.11	0.94	1.2
10	16.88	0.25	1.125	10
11	6.19	0.5	0.105	1.7

Accuracy [23] is one of the most commonly used means, which is often used in binary classification. In clustering, we only need to transform multiclassification into a two-classification problem by transforming the problem into a judgment of consistency. If the tested algorithm used is very good, the calculated value is 1. The accuracy of the evaluated algorithm can be calculated by (12).

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \quad (12)$$

Rand index [24] is judged and calculated by comparing tags with existing results. It calculates the number of the same cluster and the number of different clusters by judging whether it is a cluster. But RI has problems with random tags. At this time, adjusting RI [25] was proposed to solve this problem. Equations (13) and (14) show the calculation process of RI and ARI

$$\text{RI} = \frac{a + d}{a + b + c + d}, \quad (13)$$

$$ARI = \frac{RI - E(RI)}{\max(RI) - E(RI)}. \quad (14)$$

In view of the knowledge of the assignment of base truth classes and our prediction of the assignment of clustering algorithms for the same sample, mutual information [26] (MI) is the degree of closeness between the two is expressed in the form of joint probability. It can measure the clustering result of the clustering algorithm. It compares the joint probability density of the existing tags and the clustering algorithms. Normalized MI [27] is obtained by calculating the arithmetic mean of MI. Equations (15), (16), and (17) show the calculation process of MI and NMI.

$$H(X) = -\sum_i p(x_i) \log(x_i), \quad (15)$$

$$I(X; Y) = \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x)p(y)}, \quad (16)$$

$$NMI(X; Y) = 2 \frac{I(X; Y)}{H(X) + H(Y)}. \quad (17)$$

Completeness [28, 29] is a measure of cluster labels given ground truth. Its idea is illustrated by an example: there are three classes, and each class has a fixed arrangement of students. When all the students go back to their class, its value is the largest. But when the clustering algorithm guides the students back to the class, there will be errors, so that the students who should not be in this class will appear in this class. Then, the value will decrease.

4.2. Data and Comparison Experiments. The experimental datasets are the digits dataset, wine dataset, and heart disease (Cleveland) dataset. Through following UCI dataset experiments, we can see that it has good performance under the four evaluation indexes. The UCI dataset wine is 13-dimensional, UCI dataset heart disease is 14-dimensional, and UCI dataset digits is 64-dimensional. Experiments are compared with the original FDP, K-means, DBSCAN, and spectral clustering algorithm, through the above evaluation criteria.

For each set of datasets, we use the idea of 10-fold cross-validation. Each dataset is divided into ten parts according to the number of samples and labeled from 1 to 10. In the first experiment, the part marked 1 is removed and the remaining data is used for testing. At this time, the values of all clustering algorithms under different evaluation indexes are obtained. According to this method, 10 sets of data are obtained. For these 10 groups of data, for example, ACC of the IT-RCD-FDP part has 10 results. We remove the maximum and minimum values and then calculate the draw value as the final experimental result. For K-means algorithm, the center point needs to be randomly selected during initialization. This brings a certain amount of randomness. In order to solve this problem, we select the first k sample values as the center in the experiment. In this way, the randomness can be eliminated. After all the data are processed above, results are obtained in Tables 2–4.

TABLE 2: UCI experiments digits.

Type	ACC	NMI	ARI	Completeness
FDP	0.51	0.56	0.53	0.41
K-means	0.49	0.73	0.66	0.74
DBSCAN	0.59	0.75	0.63	0.66
Spectral	0.53	0.61	0.67	0.63
IT-RCD-FDP	0.66	0.71	0.77	0.78

TABLE 3: UCI experiments wine.

Type	ACC	NMI	ARI	Completeness
FDP	0.51	0.54	0.68	0.64
K-means	0.35	0.42	0.37	0.38
DBSCAN	0.61	0.51	0.65	0.63
Spectral	0.62	0.73	0.71	0.61
IT-RCD-FDP	0.64	0.61	0.73	0.66

TABLE 4: UCI experiments heart disease (Cleveland).

Type	ACC	NMI	ARI	Completeness
FDP	0.09	0.17	0.08	0.06
K-means	0.12	0.19	0.14	0.12
DBSCAN	0.01	0.03	0.03	0.05
Spectral	0.15	0.16	0.11	0.12
IT-RCD-FDP	0.13	0.21	0.25	0.23

Table 2 shows a set of comparison results of four different evaluations using UCI dataset digits, among which IT-RCD-FDP is our improved algorithm. The results show that ACC, ARI, and completeness are the best. In the digits dataset, the evaluation index NMI of IT-RCD-FDP is 7.2% higher than other algorithms. Table 3 also shows a set of comparison results of four different evaluations using UCI dataset wine. The results show that ACC, ARI, and completeness are the best. In the wine dataset, the evaluation index NMI of IT-RCD-FDP is 11% higher than other algorithms. According to wine and digits, the evaluation index NMI of IT-RCD-FDP is 23% higher than that of the original FDP. As can be seen from Table 4, although IT-RCD-FDP scored the highest on NMI, ARI, and completeness. But the overall level is very low. This also shows that the clustering algorithm is weak in the medical field. We hope to break through these scenes in the later work.

4.3. Application. Now, let us talk about applications. As mentioned above, clustering algorithm also has good applications in wireless sensor networks. Because the wireless sensors distributed in the space are just like every data point in the dataset. They all have their own characteristics and attributes. How to manage and control these sensors needs to understand them and analyze them. If wireless sensors are regarded as data points, clustering can be used to study and analyze them. In wireless sensor networks, the selection of the cluster head is particularly important. The cluster head is just like

FIGURE 6: Performance diagram. Clustering algorithm is applied to wireless sensor network performance demonstration.

the cluster center. In this way, the cluster head selection problem is transformed into the cluster center selection problem. The algorithm will cluster according to the centers. In many algorithms, low-energy adaptive clustering hierarchy (LEACH) algorithm [30] has the most reliable and efficient. Although LEACH performance is very good, there is also the problem that the selected cluster heads are too concentrated. To solve this problem, we introduce IT-RCD-FDP into LEACH to solve this problem. IT-RCD-FDP algorithm is used to cluster all the sensors in a wireless sensor network. Due to the adaptive nature of it, cluster heads can be automatically selected and clustered according to the cluster heads. In this way, wireless sensor networks are automatically divided into several clusters. Then, a sensor with the largest energy is selected as the cluster head in each round of the cluster. In this way, we can solve the problems of LEACH. Figure 6 shows the performance and effect of applying IT-RCD-FDP to LEACH.

5. Conclusions

This paper starts from two problems of FDP. Firstly, FDP algorithm is difficult to deal with high-dimensional data. This paper introduces the improved t-SNE algorithm. After such processing, the ability of the original algorithm to process high-dimensional data is improved. Secondly, FDP algorithm cannot select centers adaptively. In this paper, the change rate and difference are introduced to make the original algorithm select the centers adaptively. Finally, we apply the improved algorithm in WSN cluster head selection and achieve good results. The IT-RCD-FDP proposes that new way to solve high-dimensional data. Compared with the original algorithm, the algorithm finds a threshold point by a

mathematical method and improves the original manual judgment method to automatic operation by setting the range. In this way, the accuracy of the algorithm can be greatly improved. Through the improvement of t-SNE, the result of FDP is more accurate.

Data Availability

Previously reported data were used to support this study and are available at url = "http://archive.ics.uci.edu/ml." These prior studies are cited at relevant places within the text as references.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

First of all, I would like to express my gratitude to my parents for sheltering me from the wind and rain so that I can thrive. Secondly, I would like to thank Northwest Normal University. I would like to express my gratitude to my tutor for teaching and helping. Finally, I would like to express my gratitude to my classmates for lighting up my mind, when I am not happy and upset. When I encountered a bottleneck in my writing, the discussion and analysis with them gave me infinite inspiration. When I am restless, they are at my side to help me analyze problems and provide feasible suggestions. This work was supported by the Northwest Normal University under Grant (Research on Retina Image Segmentation and Aided Diagnosis Technology based on Deep Learning) 61962054.

References

- [1] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: a K-means clustering algorithm," *Journal of the Royal Statistical Society*, vol. 28, no. 1, pp. 100–108, 1979.
- [2] Y. J. Zhang, T. Oka, R. Suzuki, J. T. Ye, and Y. Iwasa, "Electrically switchable chiral light-emitting transistor," *Science*, vol. 344, no. 6185, pp. 725–728, 2014.
- [3] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD'96)*, vol. 8, pp. 226–231, München, German, 1996.
- [4] E. Nasibov and G. Ulutagay, "Robustness of density-based clustering methods with various neighborhood relations," *Fuzzy Sets and Systems*, vol. 160, no. 24, pp. 3601–3615, 2009.
- [5] K. Rohe, S. Chatterjee, and B. Yu, "Spectral clustering and the high-dimensional stochastic blockmodel," *The Annals of Statistics*, vol. 39, no. 4, pp. 1878–1915, 2011.
- [6] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: an overview," *WIREs Data Mining and Knowledge Discovery*, vol. 2, no. 1, pp. 86–97, 2012.
- [7] C. Kourouklas, O. Mangira, A. Iliopoulos, D. Chorozioglou, and E. Papadimitriou, "A study of short-term spatiotemporal clustering features of Greek seismicity," *Journal of Seismology*, vol. 24, no. 3, pp. 459–477, 2020.
- [8] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 841–847, 1991.
- [9] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.
- [10] M. Li, H. Wang, L. Yang, Y. Liang, Z. Shang, and H. Wan, "Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction," *Expert Systems with Applications*, vol. 150, p. 113277, 2020.
- [11] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.
- [12] X.-D. Wang, R.-C. Chen, Z.-Q. Zeng, C.-Q. Hong, and F. Yan, "Robust dimension reduction for clustering with local adaptive learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 657–669, 2019.
- [13] S. Wang and Y. Li, "A dynamic cluster model based on projection pursuit with its application to climate zoning," *Journal of Applied Meteorological Science*, vol. 18, no. 5, pp. 722–726, 2007.
- [14] X. Xue and Y. Wang, "Using memetic algorithm for instance coreference resolution," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 2, pp. 580–591, 2016.
- [15] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.
- [16] X. Xue and Y. Wang, "Optimizing ontology alignments through a memetic algorithm using both MatchFmeasure and unanimous improvement ratio," *Artificial Intelligence*, vol. 223, pp. 65–81, 2015.
- [17] X. Xue, C. Yang, C. Jiang, P.-W. Tsai, G. Mao, and H. Zhu, "Optimizing ontology alignment through linkage learning on entity correspondences," *Complexity*, vol. 2021, Article ID 5574732, 12 pages, 2021.
- [18] G. E. Hinton and S. T. Roweis, "Stochastic neighbor embedding," *Advances in Neural Information Processing Systems*, vol. 15, pp. 833–840, 2003.
- [19] L. van der Maaten and G. E. Hinton, "Visualizing high-dimensional data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [20] D. Liu and K. Sun, "Random forest solar power forecast based on classification optimization," *Energy*, vol. 187, p. 115940, 2019.
- [21] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [22] U. Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1650–1654, 2002.
- [23] J. S. Smith, B. T. Nebgen, R. Zubatyuk et al., "Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning," *Nature Communications*, vol. 10, no. 1, article 2903, 2019.
- [24] R. J. G. B. Campello, "A fuzzy extension of the Rand index and other related indexes for clustering and classification assessment," *Pattern Recognition Letters*, vol. 28, no. 7, pp. 833–841, 2007.
- [25] J. M. Santos and M. Embrechts, "On the use of the adjusted rand index as a metric for evaluating supervised classification," in *Artificial Neural Networks – ICANN 2009*, vol. 5769, pp. 175–184, Springer, 2009.
- [26] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *IEEE Transactions on Medical Imaging*, vol. 16, no. 2, pp. 187–198, 1997.
- [27] P. A. Estévez, M. Tesmer, C. A. Perez, and J. M. Zurada, "Normalized mutual information feature selection," *IEEE Transactions on Neural Networks*, vol. 20, no. 2, pp. 189–201, 2009.
- [28] M. K. Pakhira, S. Bandyopadhyay, and U. Maulik, "A study of some fuzzy cluster validity indices, genetic clustering and application to pixel classification," *Fuzzy Sets and Systems*, vol. 155, no. 2, pp. 191–214, 2005.
- [29] O. Arbelaitz, I. F. Gurrutxaga, J. Muguerza, J. M. Perez, and I. Perona, "An extensive comparative study of cluster validity indices," *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.
- [30] K. Lutful, N. Nidal, and T. Sheltami, "A fault-tolerant energy-efficient clustering protocol of a wireless sensor network," *Wireless Communications and Mobile Computing*, vol. 14, no. 2, pp. 175–185, 2014.