IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 22, 2019, accepted November 9, 2019, date of publication November 12, 2019,
date of current version November 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2953184

Cell Scene Division and Visualization Based on Autoencoder and K-Means Algorithm

JUN ZENG[1], JUAN WANG[1], (Student Member, IEEE), LIANG GUO[2], GUANGHUI FAN[1], KAIXUAN ZHANG[1], (Student Member, IEEE), AND GUAN GUI[1,3], (Senior Member, IEEE)

[1]College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[2]China Academy of Information and Communications Technology, Beijing 100191, China
[3]Nanjing Great Information Technology Company, Ltd., Nanjing 210046, China

Corresponding authors: Liang Guo (guoliang1@caict.ac.cn) and Guan Gui (guiguan@njupt.edu.cn)

This work was supported in part by the National Science and Technology Major Project, Ministry of Science and Technology of China, under Grant TC190A3WZ-2, in part by the National Natural Science Foundation of China under Grant 61901228, in part by the Jiangsu Specially Appointed Professor Program under Grant RK002STP16001, in part by the Summit of the Six Top Talents Program of Jiangsu under Grant XYDXX-010, in part by the Program for High-Level Entrepreneurial and Innovative Talents Introduction under Grant CZ0010617002, and in part by the 1311 Talent Plan of Nanjing University of Posts and Telecommunications.

ABSTRACT For the network service construction and optimization of wireless cell, the effective scene division is an important basis for formulating more accurate network construction schemes and optimization strategies. The traditional cell scene division method is manually divided according to the single-dimensional business indicators, but there are some problems such as the inaccuracy of division and the inability to visualize. In this paper, we propose a cell scene division and visualization method based on autoencoder and K-means algorithm. We train an autoencoder network to conduct the dimension reduction of the wireless perception key quality indicator (KQI) data of cells, and then use elbow method and K-means algorithm to cluster the dimension-reduced data precisely. Through statistical analysis and comparison of indicators of cells in different classes obtained by clustering, we finally achieve accurate cell scene division and visualization.

INDEX TERMS Scene division, autoencoder, K-means, elbow method, machine learning.

I. INTRODUCTION

With the rapid development of mobile communication network [1]–[8], the network service quality of wireless cell has gradually become a key factor in the core competitiveness of communication operators. In modern mobile communication system, wireless cell carries mixed services such as voice, data and multimedia, and different services occupy different system resources. As the number of mobile communication users and the amount of business increase, different business in the network is distributed in different wireless cells, which leads to different resource allocation requirements of wireless cells. In order to achieve fine management of cells and effective utilization of resources, it is necessary to divide the whole network into different business scenes accurately. According to the division of business scenes of each cell, we can accurately calculate the radio bearer resource needs of each cell, and develop different cell-level radio resource allocation [9], [10] and expansion strategies to avoid taking the loss in the radio resource allocation estimation, and provide a reference for the accurate expansion of wireless network resources, so as to better improve user satisfaction, and ultimately achieve the purpose of improving the market competitiveness of communication operators.

At present, the traditional cell scene division is mainly based on the single-dimension business indicator and manually divided according to the experience. This kind of method is a coarse-grained qualitative division, which cannot consider the occupancy of resources by cell business in an all-round way, so it cannot be used as an accurate basis for adjustment and optimization, nor can it guide network construction and capacity expansion adjustment. At the same time, the real number of classes cannot be given in advance when dividing scenes manually, and too many or too few classes will lead to the inaccurate scene division. In addition, the dimension of the wireless perception KQI data of cells is too high to visualize the division result.

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

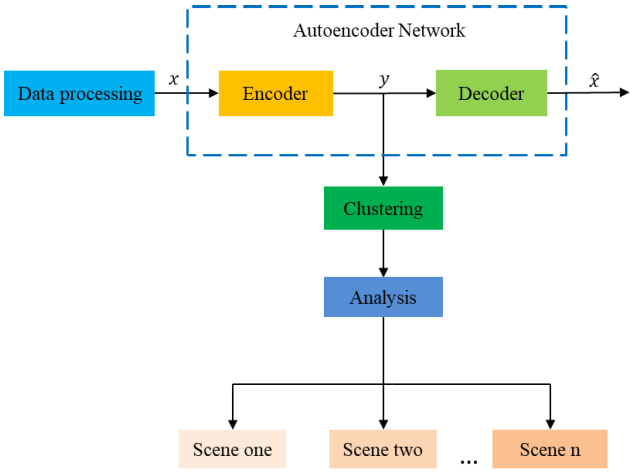


FIGURE 1. Structure of the system.

With the rise of machine learning and deep learning [11]–[20], many traditional problems in the field of mobile communications have new solutions and methods [21]–[30]. In this paper, we propose a method based on the autoencoder and K-means algorithm to achieve the cell scene division. We firstly use the autoencoder network to conduct the dimension reduction of the wireless perception KQI data of cells, and determine the real clustering number of the dimension-reduced data by the elbow method. Then, we use the K-means algorithm to cluster the dimension-reduced data. Through the statistical analysis and comparison of indicators of raw cell data in different classes obtained by clustering, we eventually achieve the division and visualization of the cell scenes.

The rest of this paper is organized as follows. In section II, we show the system model. In section III, we introduce a normalization method and the related algorithms. In section IV, we present the experimental results and conduct different comparative analysis. In section V, we conclude the whole paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

Our system model is mainly composed of data processing, autoencoder network, clustering and statistical analysis modules. Firstly, we need to process the wireless perception KQI data of cells, including data cleaning and data normalization. Secondly, we train an autoencoder network using the processed data. When the network converges, we take the output of the encoder as the dimension reduced data and visualize it. Then, we use elbow method and K-means algorithm to cluster the dimension reduced data precisely. Through the statistical analysis and comparison of indicators of raw cell data in different classes, we can achieve the cell scene division, at the same time, we can also give an explanation of the visualized image of dimension reduced data. The structure of our system is shown in Fig. 1.

The data we used in the experiment is provided by the communication operator. The wireless perception KQI data of a single cell consists of eight records called eight busy hours. Each record contains twenty-nine indicators, fourteen of which are KQI indicators, and we only use these fourteen indicators as features for experiments and analysis. For the wireless perception KQI data of each cell, there are some missing values in the data. If the sum of missing values of fourteen features in eight records exceeds 10%, we will discard the data of the current cell, otherwise, these missing values will be filled with the median of the indicators of the cell they belong to.

III. OUR PROPOSED METHOD

A. DATA NORMALIZATION

In this paper, we need to normalize the fourteen features so as to scale all the features into the same interval. Among the fourteen features, there are seven positive indicators and seven negative indicators. A positive indicator means that the larger the value, the better the indicator (e.g. page download rate), while a negative indicator means the smaller the value, the better the indicator (e.g. page display average time-delay). For the positive and negative indicators, the method called threshold normalization as shown in (1) and (2) is used for data normalization processing respectively,

$$x^* = \frac{x - x_{threshold}}{x_{max} - x_{threshold}} \tag{1}$$

$$x^* = \frac{x_{threshold} - x}{x_{threshold} - x_{min}} \tag{2}$$

where x_{max} and x_{min} represent the maximum and minimum values of feature x respectively, and $x_{threshold}$ represents the threshold of feature x, and these thresholds actually are the reference for communication operators to define whether KQIs are good or bad. For a positive indicator, if its value is below its threshold, then we think that this indicator is bad. On the contrary, if the value of a negative indicator is higher than its threshold, we think that this indicator is bad. After normalization, the values of each feature are scaled into a small interval of the same order of magnitude, which prepares the data for the following experiments.

Additionally, here we introduce the min-max normalization method and the z-score normalization method for the next comparative experiments. The min-max normalization method is defined as:

$$x^* = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3}$$

where x_{max} and x_{min} represent the maximum and minimum values of feature x respectively. By using this normalization method, the values of all indicators are scaled to [0,1]. As for the z-score normalization method, each indicator is subject to standard normal distribution after normalization by using this method, that is, the mean value is 0, and the variance is 1, and its formula is as follows:

$$x^* = \frac{x - x_{mean}}{x_{std}} \tag{4}$$

where x_{mean} represents the mean value of feature x, and x_{std} represents the standard deviation of feature x.

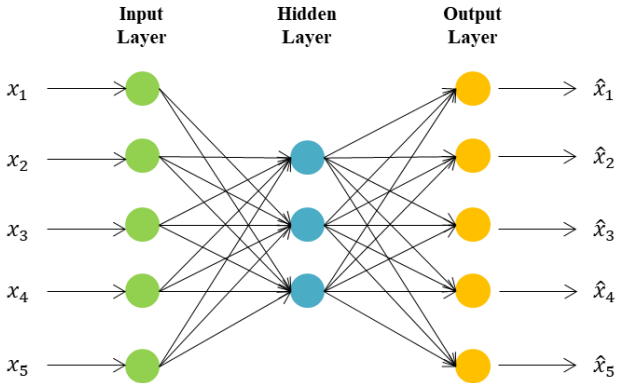


FIGURE 2. Structure of the autoencoder network.

B. AUTOENCODER NETWORK

Autoencoder is a kind of artificial neural network in deep learning, which can learn the efficient representation of input data through unsupervised learning. This efficient representation of input data is called encoding, whose dimension is generally much smaller than the input data, making the autoencoder available for dimension reduction, so it is often used in the fields of data dimension reduction, anomaly detection and so on [31]–[33]. The autoencoder network consists of symmetric encoder and decoder, and it learns the compressed representation of the input data by minimizing the reconstruction error. A basic autoencoder network containing one hidden layer is shown in Fig. 2. In practice, encoder and decoder networks may contain more than one hidden layer.

Encoder : In Fig. 2, the network between the input layer and the hidden layer is called the encoder. The activation of the i-th unit in the hidden layer is defined as:

$$h_i = a\left(\sum_{j=1}^{n} W_{ij}^{input} x_j + b^{input}\right) \tag{5}$$

where x represents the input vector ($x \in R^n$), W^{input} is the encoder weight matrix with size $d \times n$, b^{input} is the bias of hidden layer units with size $d \times 1$, a represents the activation function, and we will introduce the activation function later. In the encoder, therefore, the input vector is encoded into a low-dimensional vector h ($h \in R^d$) through the hidden layer containing d units ($d < n$).

Decoder : The network between the hidden layer and the output layer is called the decoder in Fig. 2. In the decoder, low-dimensional vector h is decoded back to the original input space R^n. The mapping function is given by:

$$\widehat{x_i} = a\left(\sum_{j=1}^{d} W_{ij}^{hidden} h_j + b^{hidden}\right) \tag{6}$$

where h represents the input low-dimensional vector, W^{hidden} is the decoder weight matrix with size $n \times d$, b^{hidden} is the bias of output layer units with size $n \times 1$.

In the whole encoding and decoding process, we only have one constraint condition on the autoencoder network, which makes the input and output of the network are as equal as possible. The loss function of the autoencoder network is defined as follows:

$$L(x, \widehat{x}) = \frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(x_j^i - \widehat{x}_j^i\right)^2 \tag{7}$$

where x^i and $\widehat{x}^i$ represent the input and output of the i-th sample in the autoencoder network respectively, they are both n-dimensional vectors, and m represents the number of samples in dataset.

After multiple iterations through forward propagation and back propagation, we force the autoencoder to learn the low-dimensional efficient representation of high-dimensional data. And then, the output of encoder is the dimension reduced data we need.

TABLE 1. Structure of the autoencoder.

Layer	Activation Function	Output Dimension
Input	/	$N \times 112$
Dense	Relu	$N \times 64$
Dense	Relu	$N \times 32$
Dense	Relu	$N \times 16$
Dense	/	$N \times 2$
Dense	Relu	$N \times 16$
Dense	Relu	$N \times 32$
Dense	Relu	$N \times 64$
Dense	/	$N \times 112$

As shown in Table 1, the autoencoder network used in this paper has nine layers including the input. For a cell, the data of the fourteen features in the eight records is a 8×14 matrix in mathematics, we expand it into a vector of 112, so the dimension of the input layer is $N \times 112$, where N represents the number of cells. For the hidden layer, we use the rectified linear unit (Relu) activation function, which is a nonlinear activation function, whose mathematical expression is shown in (8). In fact, activation function contains linear functions and nonlinear functions, but we generally use nonlinear activation functions in neural networks because it can learn more complex functions during the training of neural network.

$$Relu(x) = \begin{cases} x, if & x > 0 \\ 0, if & x \leq 0 \end{cases} \tag{8}$$

It is worth noting that we do not use activation function for the output layer units of the encoder and decoder. For the encoder, the low-dimensional features generated by its output layer units have become extremely important. When it is passed to the decoder, fewer changes can maximize the retention of the original features. For the decoder, the data produced by its output layer units should vary in the same

range as the input of the encoder. If activation function is used in the output layer units of the decoder, we cannot keep the change interval of the input and output of the autoencoder in the same range, and the training of autoencoder also loses its meaning.

C. ELBOW METHOD

Elbow method is used to determine the real clustering number of data in machine learning [34]. Its core idea is that with the increase of the clustering number k, the degree of aggregation of each cluster will gradually increase, then the sum of the squared errors (SSE) will naturally become smaller. When the k value reaches the real clustering number, the decreasing range of the SSE will decrease sharply and become flat with the increase of the k value. At this point, the relationship between the SSE and k value is similar to the shape of an elbow, and the k value corresponding to this elbow is the real clustering number of data, which is why we called it elbow method. The calculation formula of SSE is given by:

$$SSE = \sum_{i=1}^{k} \sum_{p \in c_i} |p - m_i|^2 \tag{9}$$

where C_i is the i-th class, p is the sample in C_i, m_i is the centroid of C_i, and k is the number of clusters.

D. K-MEANS ALGORITHM

K-means is an unsupervised learning algorithm. It is the most commonly used clustering algorithms because of the small iterations and the fast convergence speed. It can implement classification of data through multiple iterations and has been widely used in customer segmentation, image segmentation and other fields [35]–[37]. The algorithm flow is as follows:

1) Set the k value of clustering number based on user choice.
2) k samples are randomly selected from the dataset as the initial clustering center.
3) Traverse all samples and divide each sample into the nearest clustering center to form a class.
4) Calculate the average value of each class and take it as the new clustering center.
5) Repeat 3-4 until the k clustering centers no longer change.

The distance from samples to different clustering centers is generally measured by Euclidean distance,

$$dis(p, m_i) = |p - m_i|^2 \tag{10}$$

where p is the sample of C_i, and m_i is the centroid of C_i.

IV. EXPERIMENT RESULTS

A. ANALYSIS OF CLUSTERING RESULT OF DIMENSION REDUCED DATA

In this section, dimension reduced data is used to conduct the clustering and analysis to achieve the scene division. We firstly use the normalized data to train an autoencoder network as shown in Table 1. After many iterations, when the network converges, the output of the encoder is the 2D features we need. Then, we use the low-dimension features to conduct the clustering process. Fig. 3 shows the relationship between the different k values and SSE when we cluster the 2D features using the K-means algorithm.

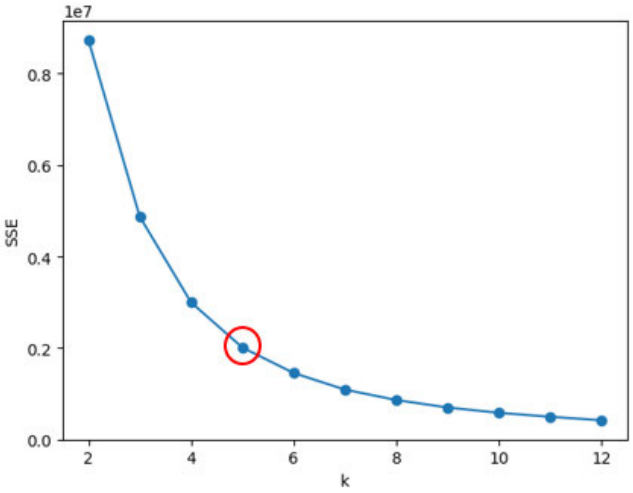


FIGURE 3. Relationship between the class k and SSE.

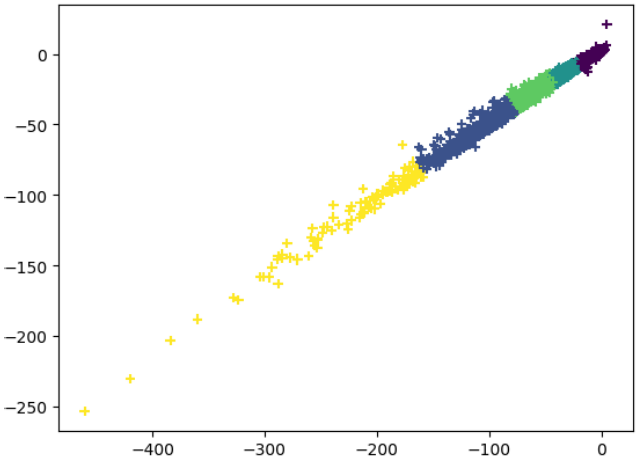


FIGURE 4. Result of the clustering.

Through observation, we find the SSE decreases sharply with the increase of the k value. When the k value is greater than five, the decreasing range of the SSE tends to be flat. And then, according to the idea of the elbow method, we determine that the real clustering number of the 2D features is five. The clustering result is shown in Fig. 4.

After the clustering, we conduct statistical analysis on the raw cell data (unnormalized data, including twenty-nine indicators) of these five classes. We separate the raw cell data by class, and then calculate the average values of various indicators in different classes of cells. Here, for the sake of convenience, we only select ten out of twenty-nine indicators for comparison and analysis, at the same time, we replace the names of indicators with numbers for representation, and the corresponding relationship between the indicators and the numbers is shown in the Table 2.

TABLE 2. Correspondence between indicators and numbers.

Indicators	Numbers
Number of page requests(times)	1
Number of video playing requests(times)	2
Number of instant messaging service requests(times)	3
Number of game service requests (times)	4
Page display average time-delay(ms)	5
Average cache time-delay for video playing(ms)	6
Instant messaging response time-delay(ms)	7
Number of video playing pauses(times)	8
Page download rate(kbps)	9
Video download rate(kbps)	10

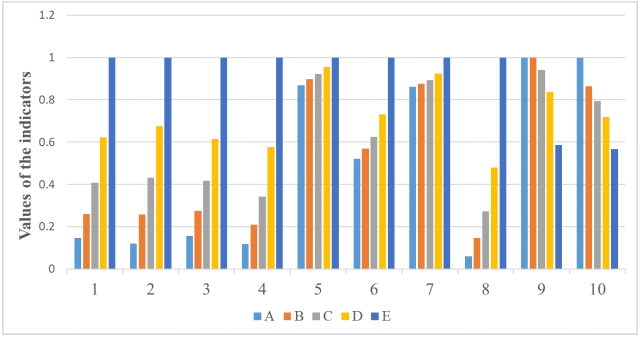


FIGURE 5. Comparison of indicators of cells in different classes.

Fig. 5 shows the statistical result, as can be seen from the figure, the abscissa represents the number corresponding to the indicator, while the ordinate represents the normalized value of the indicator. The five classes of cells obtained by clustering are named as class A, B, C, D and E respectively. We can find that from class A to class E, the values of various request times indicators (indicator 1 to 4) are gradually increasing, the values of different time-delay indicators (indicator 5 to 7) are gradually increasing, the number of video playing pauses (indicator 8) is gradually increasing, and the download rates of indicator 9 and 10 are gradually decreasing. Therefore, according to the change trends of these indicators, the cells of class A to class E are defined as very low business, low business, normal business, high business, and very high business, and then we achieve the scene division of cells.

In addition, here we give an explanation of the 2D features obtained by dimension reduction. If the horizontal and vertical coordinate values of the 2D data are smaller, that is to say, the closer the position of the data point is to the lower left in Fig. 4, the higher the business of the cell represented by the data point is. For cells with low business, we can appropriately configure them with less radio resources to prevent

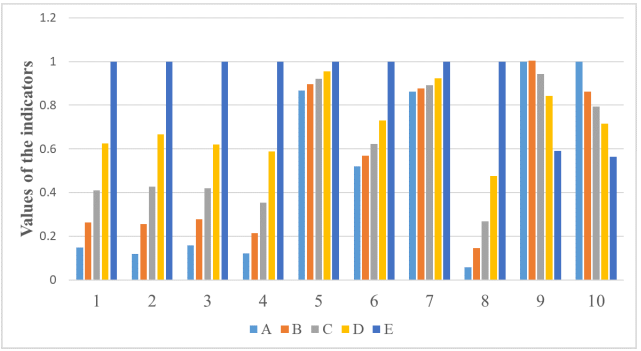


FIGURE 6. Comparison of indicators of cells in different classes.

waste of resources and rise of costs. By contrast, for cells with high business, we should allocate more radio resources to them to prevent network congestion from causing user complaints, so as to further achieve the full utilization of radio resources and the refined management of cells.

B. ANALYSIS OF CLUSTERING RESULT OF RAW DATA

In this section, the raw features of cells, instead of the 2D features, are used to conduct the clustering and analysis to achieve the scene division. For the processed data, we remove the dimension reduction process and directly cluster the 112D features. Here, for the comparison, we still cluster the data into five classes. After the clustering, we separate the raw cell data by class, and then calculate the average values of various indicators in different classes of cells. It should be pointed out that we cannot display the clustering result duo to the high dimension of raw cell data, but only the statistical result as shown in Fig. 6.

As can be seen from Fig. 6, from class A to class E, the distribution the indicators in different classes of cells is almost the same as the result in Fig. 5, and these change trends mentioned above also exist, for example, the values of various request times indicators are gradually increasing, the values of different time-delay indicators are gradually increasing and so on.

Fig. 7 shows the number of cells in different classes obtained by clustering the raw 112D features and the dimension-reduced 2D features. It can be seen from Fig. 7 that after clustering with 112D features, the number of cells in each class is basically the same as the result of clustering 2D data.

From the analysis of Fig. 6 and Fig. 7, it shows that the features after dimension reduction basically retain the information of the raw features, which proves that dimension reduction for visualization is effective, and the 2D features can well represent the characteristics of cells.

C. DIMENSION REDUCTION BASED ON MIN-MAX NORMALIZATION

In this section, we change the normalization method of raw cell data for dimension reduction and clustering analysis. Instead of the threshold normalization method as shown in

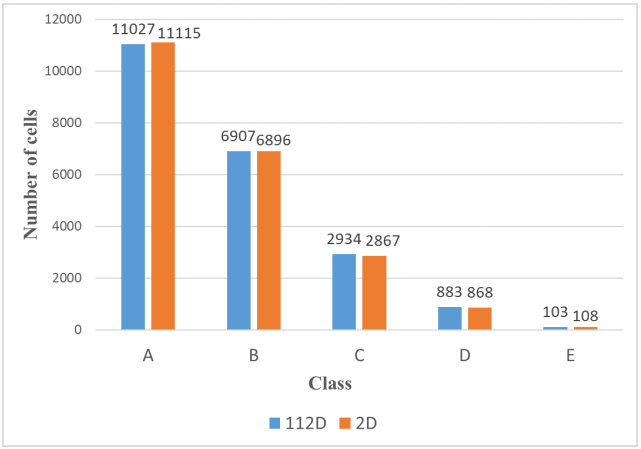


FIGURE 7. Number of cells in different classes.

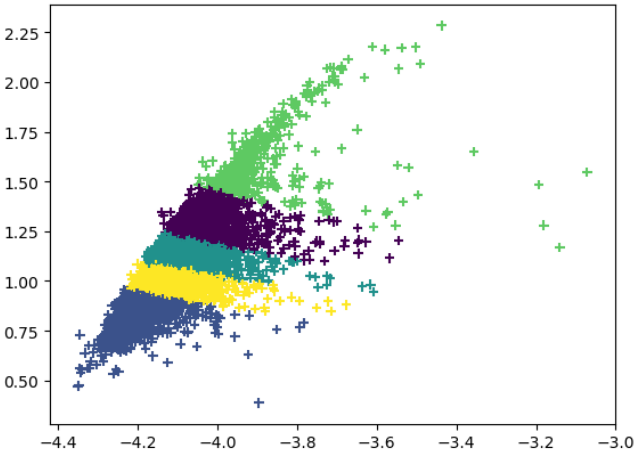


FIGURE 8. Result of the clustering.

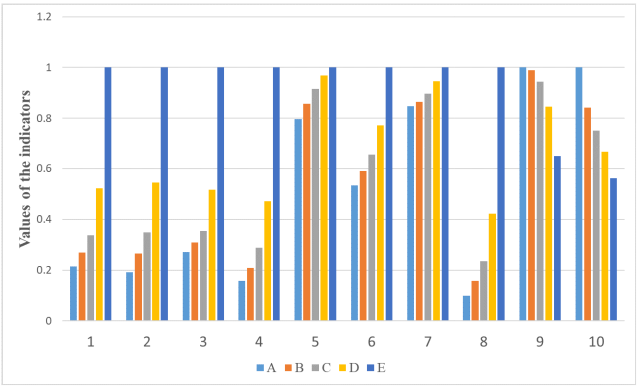


FIGURE 9. Comparison of indicators of cells in different classes.

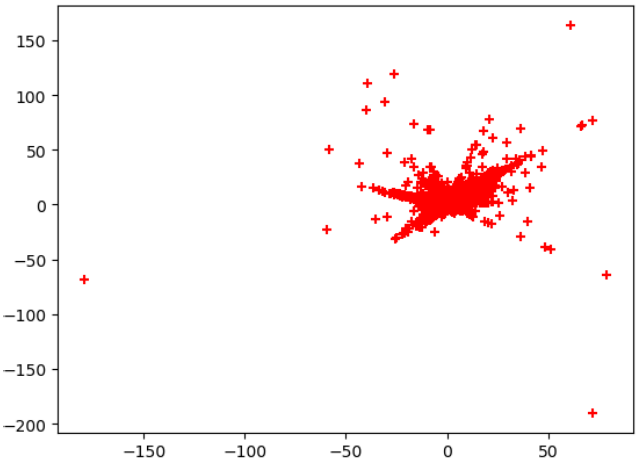


FIGURE 10. Result of dimension reduction.

(1) and (2), we use the min-max method to normalize the raw cell data, and then the processed data is put into autoencoder network for training. When the network converges, we take the 2D data of encoder and cluster it. The clustering result is shown in Fig. 8.

As can be seen from Fig. 8, we still cluster the dimension reduced 2D features into five classes, and then we calculate the average values of various indicators of raw cell data in these five classes. The statistical result is shown in Fig. 9.

From the observation of Fig. 9, we can find that from class A to class E, the values of various request times indicators are gradually increasing, the values of different time-delay indicators are gradually increasing, the number of video playing pauses is gradually increasing, and the download rates of indicator 9 and 10 are gradually decreasing. However, there is a big difference between Fig. 5 and Fig. 9. In Fig. 5, the values of these ten indicators are gradually increasing or decreasing, in other words, the values of these indicators change more evenly. But in Fig. 9, for example, the values of indicator 1 to 4 are gradually increasing from class A to class C, nevertheless, in the process from class C to class D and class D to class E, there is a mutation in the values of indicators 1 to 4, whose values are far greater than those in class A, B and C, which also exists in the analysis of other indicators.

From the analysis of the above two figure, we can find that when we use the threshold normalization method we designed to divide the scene, the division of classes will be more uniform, which shows the superiority of the threshold normalization method in scene division of cells.

D. DIMENSION REDUCTION BASED ON Z-SCORE NORMALIZATION

In this section, we choose z-score method to normalize the raw cell data, and then we perform the same operation on the data as described in the previous section. The result of dimension reduction is shown in Fig. 10.

From Fig. 10 we can see that the data distribution after dimension reduction is basically concentrated on the vicinity of the origin of the coordinate axes, which formed a very dense cluster. At this time, no matter whether we use the K-means algorithm or other clustering algorithms, we cannot cluster the 2D data well, and then we cannot divide the scene,

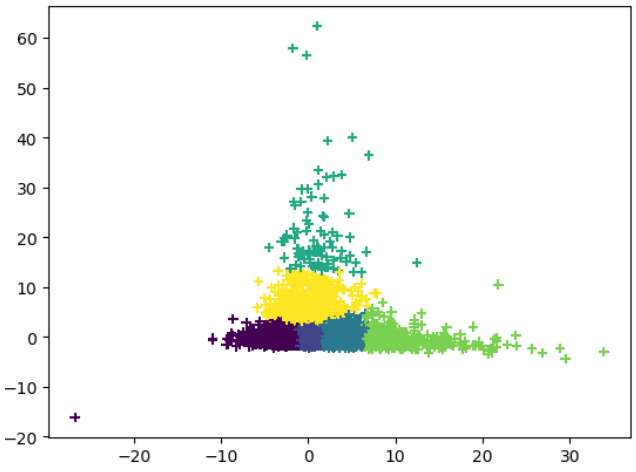


FIGURE 11. Result of dimension reduction.

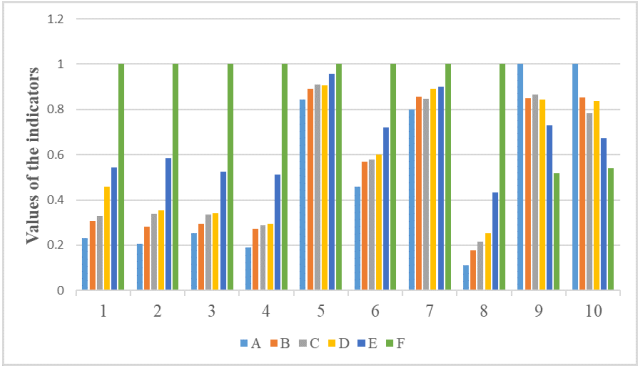


FIGURE 12. Comparison of indicators of cells in different classes.

which means that the z-score normalization method is not a good choice in the case of cell scene division.

E. DIMENSION REDUCTION BASED ON PCA ALGORITHM

In this section, we change the dimension reduction algorithm. Instead of the autoencoder network, we choose the PCA algorithm to perform dimension reduction of the raw cell data. After data processing, PCA algorithm is used to reduce the dimension of data to 2D, then elbow method and K-means algorithm are used to cluster the dimension-reduced data precisely. The clustering result can be seen in Fig. 11.

Fig. 11 shows that the 2D features are clustered to six classes by using elbow method. As before, we conduct statistical analysis on the raw cell data of these six classes. The statistical result is shown in Fig. 12.

We can observe six classes of cells in Fig. 11. The difference of the average values of indicators in some classes is very small (e.g. indicator 1 to 4 in class C and class D), while the difference of the average value of indicators in other classes is very large (e.g. indicator 1 to 4 in class E and class F). At the same time, the average values of some indicators are not decreasing gradually from class A to class F. For example, the average value of indicator 10 firstly decreases gradually from class A to class C, and then increases from class C to class D, finally it decreases from class D to class F.

From the above analysis, we can get that our clustering result is very poor, which makes it impossible to divide the cells uniformly. Meanwhile, we cannot give a good explanation of the distribution of data points in Fig. 12. The reason, we believe that the PCA is a linear dimension reduction algorithm, which can only reduce the linear components between the dimensions in the data. On the contrary, the autoencoder can learn the nonlinear relationship between the dimensions in the data, so it can achieve better dimension reduction, which shows the advantage of autoencoder in dimension reduction of wireless cell data. As a result, we can get different classes of cells with more uniform division after clustering.

V. CONCLUSION

In this paper, we have proposed a scene division and visualization method based on the autoencoder and K-means algorithm. Through conducting dimension reduction and clustering analysis on the raw cell data, we achieved the scene division of cells. In addition, we made different comparisons during experiment, which proved the effectiveness of dimension reduction of autoencoder and the superiority of threshold normalization method in cell scene division. We firmly believe that the cell scene division enables communication operators to develop more elaborate network construction schemes and optimization strategies, which plays an extremely important role in improving the quality of cell network services and enhancing the core competitiveness of communication operators.

ACKNOWLEDGMENT

The authors would like to express their thanks to the BOCO Inter-Telecom for their data.

REFERENCES

[1] Z. Zhou, Y. Guo, Y. He, X. Zhao, and W. M. Bazzi, "Access control and resource allocation for M2M communications in industrial automation," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 3093–3103, May 2019.

[2] M. Liu, J. Yang, and G. Gui, "DSF-NOMA: UAV-assisted emergency communication technology in a heterogeneous Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5508–5519, Jun. 2019, doi: 10.1109/JIOT.2019.2903165.

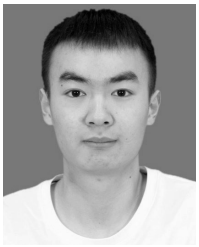
[3] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato, "An intelligent traffic load prediction-based adaptive channel assignment algorithm in SDN-IoT: A deep learning approach," *IEEE Internet Things Journal*, vol. 5, no. 6, pp. 5141–5154, Dec. 2018.

[4] B. Mao, Z. M. Fadlullah, F. Tang, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.

[5] Z. Zhou, M. Dong, K. Ota, G. Wang, and L. T. Yang, "Energy-efficient resource allocation for D2D communications underlaying cloud-RAN-based LTE-A networks," *IEEE Internet Things J.*, vol. 3, no. 3, pp. 428–438, Jun. 2016.

[6] G. Gui, H. Sari, and E. Biglieri, "A new definition of fairness for non-orthogonal multiple access," *IEEE Commun. Lett.*, vol. 23, no. 7, pp. 1267–1271, May 2019.

[7] M. Wei, S. Sezginer, G. Gui, and H. Sari, "Bridging spatial modulation with spatial multiplexing: Frequency-domain ESM," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 6, pp. 1326–1335, Oct. 2019.
[8] Z. Zhou, J. Gong, Y. He, and Y. Zhang, "Software defined machine-to-machine communication for smart energy management," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 52–60, Oct. 2017.
[9] M. Liu, J. Yang, T. Song, J. Hu, and G. Gui, "Deep learning-inspired message passing algorithm for efficient resource allocation in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 641–653, Jan. 2019.
[10] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA based heterogeneous IoT with imperfect SIC," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2885–2894, Apr. 2019.
[11] X. Sun, G. Gui, Y. Li, R. P. Liu, and Y. An, "ResInNet: A novel deep neural network with feature reuse for Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 679–691, Feb. 2019.
[12] X. Sun, Y. Li, G. Gui, and H. Sari, "Echo-state restricted Boltzmann machines: A perspective on information compensation," *IEEE Access*, vol. 7, no. 1, pp. 16281–16290, Jan. 2019.
[13] T. Zhou, S. Yang, L. Wang, J. Yao, and G. Gui, "Improved cross-label suppression dictionary learning for face recognition," *IEEE Access*, vol. 6, pp. 48716–48725, Aug. 2018.
[14] L. Shao, M. Li, L. Yuan, and G. Gui, "InMAS: Deep learning for designing intelligent making system," *IEEE Access*, vol. 7, no. 1, pp. 51104–51111, 2019.
[15] J. Pan, Y. Yin, J. Xiong, W. Luo, G. Gui, and H. Sari, "Deep learning-based unmanned surveillance systems for observing water levels," *IEEE Access*, vol. 6, no. 1, pp. 73561–73571, Dec. 2018.
[16] Y. Zhao, Q. Chen, W. Cao, J. Yang, J. Xiong, and G. Gui, "Deep learning for risk detection and trajectory tracking at construction sites," *IEEE Access*, vol. 7, no. 1, pp. 30905–30912, 2019.
[17] W. Li, H. Liu, Y. Wang, Z. Li, Y. Jia, and G. Gui, "Deep learning-based classification methods for remote sensing images in urban built-up areas," *IEEE Access*, vol. 7, no. 1, pp. 36274–36284, 2019.
[18] W. Hua, F. Dai, L. Huang, J. Xiong, and G. Gui, "HERO: Human emotions recognition for realizing intelligent Internet of Things," *IEEE Access*, vol. 7, no. 1, pp. 24321–24332, 2019.
[19] Y. Sun, X. Mao, S. Hong, W. Xu, and G. Gui, "Template matching-based method for intelligent invoice information identification," *IEEE Access*, vol. 7, no. 1, pp. 28392–28401, 2019.
[20] H. Huang, Y. Peng, J. Yang, W. Xia, and G. Gui, "Fast beamforming design via deep learning," *IEEE Trans. Veh. Technol.*, to be published, doi: 10.1109/TVT.2019.2949122.
[21] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.
[22] H. Huang, S. Guo, G. Gui, Z. Yang, J. Zhang, H. Sari, and F. Adachi, "Deep learning for physical-layer 5G wireless techniques: Opportunities, challenges and solutions," *IEEE Wireless Commun. Mag.*, to be published, doi: 10.1109/MWC.2019.1900027.
[23] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep-learning-based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3027–3032, Mar. 2019.
[24] J. Sun, W. Shi, Z. Yang, J. Yang, and G. Gui, "Behavioral modeling and linearization of wideband RF power amplifiers using BiLSTM networks for 5G wireless systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 10348–10356, Nov. 2019, doi: 10.1109/TVT.2019.2925562.
[25] J. Wang, Y. Ding, S. Bian, Y. Peng, M. Liu, and G. Gui, "UL-CSI data driven deep learning for predicting DL-CSI in cellular FDD systems," *IEEE Access*, vol. 7, no. 1, pp. 1–10, 2019.
[26] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.
[27] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.
[28] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised learning-based fast beamforming design for downlink MIMO," *IEEE Access*, vol. 7, pp. 7599–7605, Dec. 2018.
[29] Y. Li, X. Cheng, and G. Gui, "Co-robust-ADMM-net: Joint ADMM framework and DNN for robust sparse composite regularization," *IEEE Access*, vol. 6, pp. 47943–47952, 2018.
[30] G. Gui, Y. Wang, and H. Huang, "Deep learning based physical layer wireless communication techniques: Opportunities and challenges," *J. Commun.*, vol. 40, no. 2, pp. 19–23, Feb. 2019.
[31] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
[32] Z. Chen, C. K. Yeo, B. S. Lee, and C. T. Lau, "Autoencoder-based network anomaly detection," in *Proc. Wireless Telecommun. Symp. (WTS)*, 2018, pp. 1–5.
[33] R. C. Aygun and A. G. Yavuz, "Network anomaly detection with stochastically improved autoencoder based models," in *Proc. IEEE Int. Conf. Cyber. Secur. Cloud Comput. (CSCloud)*, Jun. 2017, pp. 193–198.
[34] D. Marutho, S. H. Handaka, and E. Wijay, "The determination of cluster number at k-mean using elbow method and purity evaluation on headline news," in *Proc. Int. Seminar Appl. Technol. Inf. Commun.*, 2018, pp. 533–538.
[35] L. Ye, C. Qiu-Ru, X. Hai-Xu, L. Yi-Jun, and Y. Zhi-Min, "Telecom customer segmentation with K-means clustering," in *Proc. Int. Conf. Comput. Sci. Educ. (ICCSE)*, 2012, pp. 648–651.
[36] C. Cheng, X. Cheng, M. Yuan, C. Song, L. Xu, H. Ye, and T. Zhang, "A novel cluster algorithm for telecom customer segmentation," in *Proc. Int. Symp. Commun. Inf. Technol. (ISCIT)*, 2016, pp. 324–329.
[37] P. Shanmugavadivu and A. Shanthasheela, "K-Means clustering based bi-level coarse image segmentation," in *Proc. Int. Conf. Commun. Comput. Intell. (INCOCCI)*, 2010, pp. 254–259.



JUN ZENG is currently pursuing the master's degree in communication and information engineering with the Nanjing University of Posts and Telecommunications, Nanjing, China. His research interest includes machine learning for wireless communications.

JUAN WANG (S'19) is currently pursuing the master's degree in communication and information engineering with the Nanjing University of Posts and Telecommunications, Nanjing, China. Her research interest includes machine learning for wireless communications.

LIANG GUO is currently the Deputy Director of the Data Center Research Department, Institute of Cloud Computing and Big Data, China Academy of Information and Communications Technology. He is also a Senior Engineer, the Co-Chairman of the Ad Hoc Group of Industrial Data Center of Industrial Internet Industry Alliance, and the Head of the Working Group on New Technologies and Testing of Open Data Center Committee. He has been engaged in policy support, technical research and standard-setting in data center, networks, and IT.

GUANGHUI FAN is currently pursuing the master's degree in communication and information engineering with the Nanjing University of Posts and Telecommunications, Nanjing, China. His research interest includes machine learning for wireless communications.

KAIXUAN ZHANG (S'18) is currently pursuing the master's degree in communication and information engineering with the Nanjing University of Posts and Telecommunications, Nanjing, China. His research interest includes machine learning for wireless communications.

GUAN GUI (M'11–SM'17) received the Dr.Eng. degree in information and communication engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2012.

From October 2009 to March 2012, with the financial supported from the China Scholarship Council (CSC) and the Global Center of Education (ECOE) of Tohoku University, he joined the Department of Communications Engineering, Graduate School of Engineering, Tohoku University, as an Research Assistant and a Postdoctoral Research Fellow, respectively. From September 2012 to March 2014, he was supported by the Japan Society for the Promotion of Science (JSPS) Fellowship, as a Postdoctoral Research Fellow with Tohoku University. From April 2014 to October 2015, he was an Assistant Professor with the Department of Electronics and Information System, Akita Prefectural University. Since November 2015, he has been a Professor with the Nanjing University of Posts and Telecommunications, Nanjing, China, and also he is a Chief Scientist with Nanjing Great Information Technology Company, Ltd., Nanjing. He has published more than 200 international peer-reviewed journal/conference papers. He is currently engaged in the research of deep learning, compressive sensing, and advanced wireless techniques.

Dr. Gui received the Member and Global Activities Contributions Award from the IEEE ComSoc and the eight best paper awards, i.e., ICEICT 2019, CSPS 2019, ADHIP 2018, CSPS 2018, ICNC 2018, ICC 2017, ICC 2014, and VTC 2014-Spring. He was also selected as a Jiangsu Specially Appointed Professor, in 2016, the Jiangsu High-Level Innovation and Entrepreneurial Talent, in 2016, the Jiangsu Six Top Talent, in 2018, and the Nanjing Youth Award, in 2018. He was an Editor of the *Security and Communication Networks*, from 2012 to 2016. He has been an Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, since 2017, IEEE ACCESS, since 2018, the *KSII Transactions on Internet and Information Systems*, since 2017, and the *Journal on Communications*, since 2019, and the Editor-in-Chief of the *EAI Transactions on Artificial Intelligence*, since 2018.