IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 12, 2017, accepted October 1, 2017, date of publication October 4, 2017, date of current version March 16, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2759509

A High-Order Clustering Algorithm Based on Dropout Deep Learning for Heterogeneous Data in Cyber-Physical-Social Systems

FANYU BU
Department of Biomedical Informatics, Inner Mongolia University of Finance and Economics, Hohhot 010070, China
bufanyu@imufe.edu.cn

This was work supported in part by the National Natural Science Foundation of China under Grant 61762068, in part by the Natural Science Foundation of the Inner Mongolia Autonomous Region Grant 2017MS0610, and in part by the Higher Educational Scientific Research Projects of Inner Mongolia Autonomous Region under Grant NJZY16145.

ABSTRACT An explosive growth of cyber-physical-social systems has been witnessed owing to the wide use of various mobile devices recently. A large volume of heterogeneous data has been collected from cyber-physical-social systems in the past few years. Each object in the heterogeneous dataset is typically multi-modal, posing a remarkable challenge on heterogeneous data clustering. In this paper, we propose a high-order k-means algorithm based on the dropout deep learning model for clustering heterogeneous objects in cyber-physical-social systems. We first build three dropout stacked auto-encoders, each with three hidden layers to learn the features for the different modalities of each object. Furthermore, we establish a feature tensor for each object by using the vector outer product to fuse the learned features. At last, we devise a tensor k-means algorithm to cluster the heterogeneous objects based on the tensor distance. We evaluate the proposed high-order k-means algorithm on two representative heterogeneous data sets and results imply that the proposed high-order k-means algorithm can achieve more accurate clustering results than other heterogeneous data clustering methods.

INDEX TERMS Cyber-physical-social systems, dropout deep learning model, heterogeneous data, high-order clustering.

I. INTRODUCTION

Rcently cyber-physical-social systems (CPSS) have achieved a great progress with the broad use of mobile physical devices and social media [1]. Specially, CPSS can be viewed as an extension of the Internet of Things. Internet of Things integrates the physical devices to the information or cyber space via wireless communication networks and the Internet [2], [3]. Thus, Internet of Things is usually viewed as a kind of cyber-physical systems, as presented in Fig. 1.

Cyber-physical-social systems introduce the humans social behaviors and intelligence into cyber-physical systems by mobile personal computing and advanced communication techniques, as presented in Fig. 2.

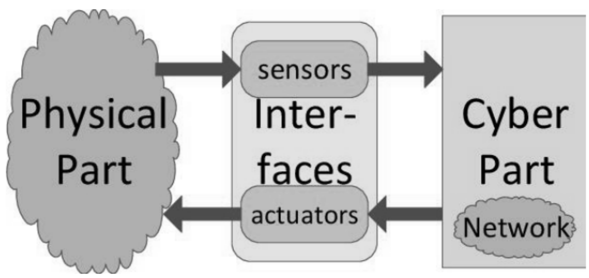

FIGURE 1. Cyber-physical systems.

From the data analyzing and processing point of view, a large volume of data, sometimes called big data, is collected from the physical world and the social world using the mobile devices such as sensors, camera, RFID and so on. Afterwards, the collected data is transmitted to the cyber space via the wireless sensor networks and the Internet. In the cyber space, the collected data is analyzed and processed to provide services such as intelligent decisions for humans. Therefore, data mining and analytics are vital for Internet of Things and cyber-physical-social systems to offer services [4].

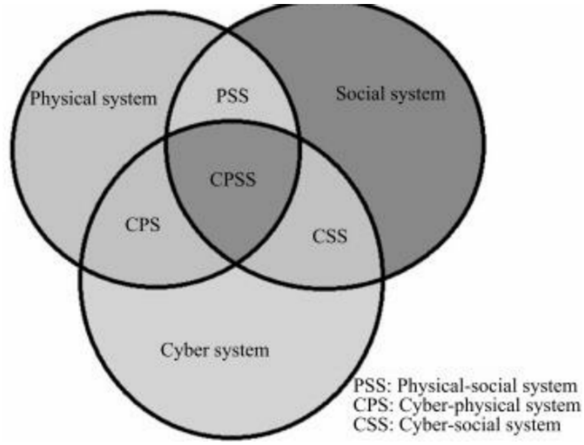

FIGURE 2. Cyber-physical-social systems.

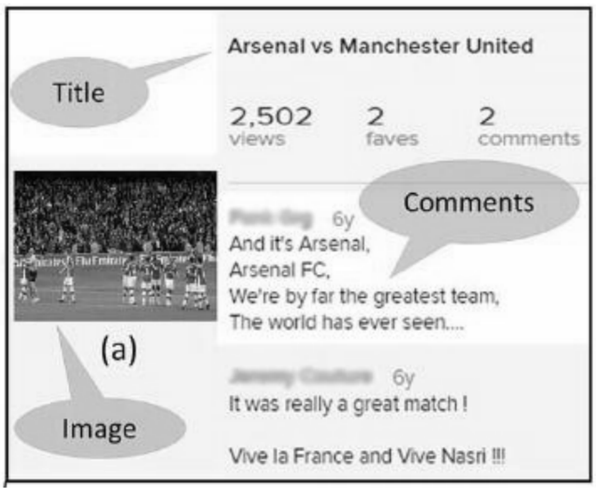

FIGURE 3. One example of heterogeneous data.

In recent years, a large number of heterogeneous objects are collected from cyber-physical-social systems. For example, with the dramatic progress of information collection techniques, various heterogeneous data is sampled from medical domain that is a typical application of cyber-physical-social systems including X-ray and CT and so on. As a result, the size of heterogeneous data is in the constant growth. Typically, heterogeneous data has the multi-modal property. Specially, each object in the heterogeneous dataset contains multiple modalities, such as texts and images. For example, the object, as shown in Fig. 3, is a representative heterogeneous sample, including one image and several paragraphs of texts (e.g., title and comments) [5].

The objects with the multi-modal property pose a remarkable challenge on the data mining for heterogeneous datasets. Specially, different modalities express distinguished information and they have complicated correlations. The example in Fig. 3 utilizes the image show the vivid scenes in the contest and utilizes the texts to display the hidden information such as the name of the contest and the comments.

In this paper, we focus on the clustering for heterogeneous data. Clustering, as one of the most vital data mining techniques, partitions the objects into different groups [6]. The objects in the same cluster share as much similarity as possible while the objects in the different groups are as different as possible. In the past decades, a lot of clustering techniques have been developed by researchers and engineers, which could be typically grouped into two classes, i.e., hard clustering and soft clustering [7]. In the hard clustering, each object is assigned into only one group while the soft clustering assigns each object into several groups. The representative hard clustering algorithms include k-means, affinity propagation and density-based clustering algorithms while the fuzzy c-means algorithm and the possibilistic c-means algorithm are two typical soft clustering techniques.

Recently, with the continue growth of heterogeneous data, heterogeneous data clustering has attracted much attention from researchers and engineers and many algorithms have been designed for clustering heterogeneous data [8]. For example, a multi-modal spectral clustering algorithm was developed by devising an objective function to extract the features for clustering the heterogeneous data. Gao and Long applied the graph theory to the heterogeneous data clustering [10], [11]. Specially, they converted the heterogeneous data clustering into a task of graph partitioning. Typical methods of this type is spectral relational clustering and consistent isoperimetric high-order co-clustering [12]. In addition, Chen *et al.* [13] proposed a co-clustering framework for heterogeneous data by applying the non-negative matrix decomposition. A representative example of the heterogeneous data clustering methods is supervised non-negative matrix decomposition that captures the correlations between every object and the clustering centers by designing a semantic space. These approaches have made some progress for heterogeneous data clustering. However, they are generally of high computational complexity and low accuracy. Zhang *et al.* [14], [15] introduced a high-order possibilistic clustering algorithm (HOPCM) based on auto-encoders for heterogeneous data. Although HOPCM outperforms other representative heterogeneous data clustering algorithm in clustering accuracy and efficiency, it still has several drawbacks. First, this method could not extract the hierarchical features for the different modalities of each object since it uses the auto-encoder with only one hidden layer to learn the features, decreasing the final clustering accuracy. Second, HOPCM has a high computational complexity since it uses the possibilistic c-means algorithm to cluster the feature tensors.

Aiming at this problems, we propose a high-order k-means algorithm (HOK-Means) based on dropout deep learning models for heterogeneous data clustering in cyber-physical-social systems. HOK-means works in three steps, i.e., feature learning, feature fusion and clustering. In the first step, we build different stacked auto-encoders [16], [17] to learn the features for different modalities of each heterogeneous object. A large number of experiments demonstrate that the stacked auto-encoders with three hidden layers performs best for feature learning [18], [19]. In addition, the dropout can reduce the over-fitting by setting the units in the hidden layers of the auto-encoder model to 0 with the possibility of 0.5 [20]. Therefore, we build different dropout stacked auto-encoders, each with three hidden layers, to learn the features for each object. In the second step, we establish a feature tensor for each object by using the vector outer product to fuse the learned features. K-means is the most widely used clustering algorithm in many application domains such as electronic business and industry production due to its high accuracy and low computational complexity. Therefore, in the last step, we design a tensor k-means algorithm to cluster the heterogeneous objects represented by the feature tensors for the final pattern. In the tensor k-means algorithm, we utilize the tensor distance to measure the similarity of each two objects in the tensor space [21]. Finally, we assess the high-order k-means algorithm based on the hybrid deep learning models on two typical heterogeneous datasets collected from cyber-physical-social systems, i.e., NUS-WIDE [22] and CUAVE [23]. Furthermore, we compare our proposed approach with the high-order possibilistic c-means algorithm in the clustering accuracy and efficiency. Results demonstrate that our algorithm outperforms the high-order possibilistic c-means algorithm.

In summary, this paper has three major contributions:

- We propose a uniform architecture that combines the deep learning with the k-means algorithm for heterogeneous data clustering. Specially, the scheme works in three steps, i.e., feature learning, feature fusion and high-order cluster;
- We design a tensor k-means algorithm for high-order feature tensors clustering by extending the k-means scheme to the tensor space. In the tensor k-means algorithm, we utilize the tensor distance to measure the similarity of each two objects in the tensor space;
- We validate the high-order k-means scheme based on the deep learning models on two heterogeneous datasets, i.e., NUS-WIDE and CUAVE. Furthermore, we compare our proposed approach with the high-order possibilistic c-means algorithm in the clustering accuracy and efficiency.

II. PROBLEM STATEMENT

Suppose that $X = \{x_1, x_2, \ldots, x_n\}$ denotes a heterogeneous dataset collected from cyber-pyhical-social systems. Each object x_i in this dataset is a multi-modal sample, that means x_i contains at least two modalities. Typically, x_i contains two modalities such as image modality and text modality or three modalities including image modality, text modality and audio modality. For example, a piece of video usually contains a set of images, some texts and a piece of audio. Considering the dataset X, the task of the high-order k-means algorithm based on the dropoout deep learning model is to cluster it into k subsets $X = X_1 \cup X_2 \cup \cdots \cup X_k$ under the condition $X_1 \cap X_2 \cap \cdots \cap X_k = \emptyset$ depending on the similarity between each two objects. There are usually many remarkable challenges for heterogeneous data clustering since each object is multi-modal. In the part, we illustrate three major problems as following.

- Feature Learning. Feature learning is the first and the fundamental stage for heterogeneous data clustering. It plays very key role on the clustering accuracy. In the past few years, many feature learning techniques have been developed, especially deep learning models. Deep learning models could extract multiple levels of features for objects by stacking some basic machine learning models [24]. For example, the deep belief networks, one of the most widely utilized deep learning models, are established by stacking some restricted Bolzmann machines [25]. Deep learning models have achieved the best performance for feature learning in these years. However, most of current deep learning models focus on feature learning for the supervised objects, so they are difficult to be applied to the clustering. Therefore, how to combine the deep learning models with the clustering algorithm for heterogeneous data is the first problem to be addressed;
- Feature Fusion. In the feature learning stage, we extract the features for each modality of every object to build several feature vectors. For example, considering a heterogeneous object with three modalities, typically including image modality, text modality and audio modality, we will form three feature vectors, i.e., image feature vector, text feature vector and audio feature vector, for the three modalities by using three deep learning models. So, the second problem is how to capture the correlations for three modalities to reveal the feature for each object by fusing the learned feature vectors.;
- Tensor Clustering. In the feature fusion stage, we apply the vector outer product for the learned feature vectors fusion and so we will get a feature tensor for each object. The final result can be obtained by clustering the heterogeneous objects represented by the feature tensors. Although a large number of clustering techniques such as k-means and affinity propagation have been devised in these years, most of them focus on the vector clustering. In other words, they could not work in the tensor mode since they cannot compute the distance between two objects in the tensor space. Therefore, how to cluster the heterogeneous objects in the tensor space is the third problem.

III. HIGH-ORDER K-MEANS BASED ON DEEP LEARNING

Fig. 4 shows the architecture of the proposed high-order k-means algorithm based on the dropout deep learning models. In detail, the proposed architecture includes three parts from bottom to top, i.e., feature learning, feature fusion and tensor clustering.

In the first part, we build three dropout stacked auto-encoders, each with three hidden layers, to learn the features for the modalities of each object, respectively. After learning

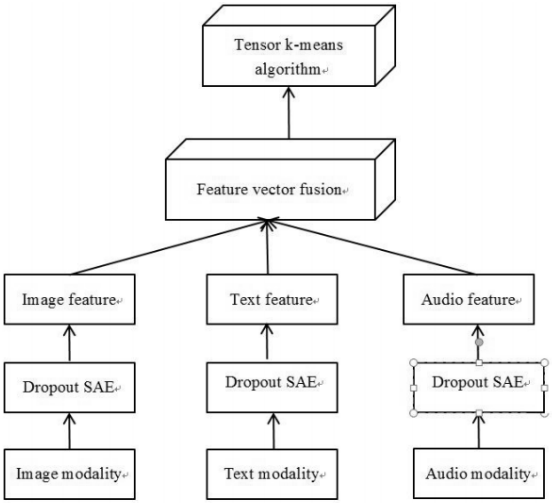

FIGURE 4. The architecture of the high-order k-means algorithm with three parts, i.e., feature learning, feature fusion and tensor clustering.

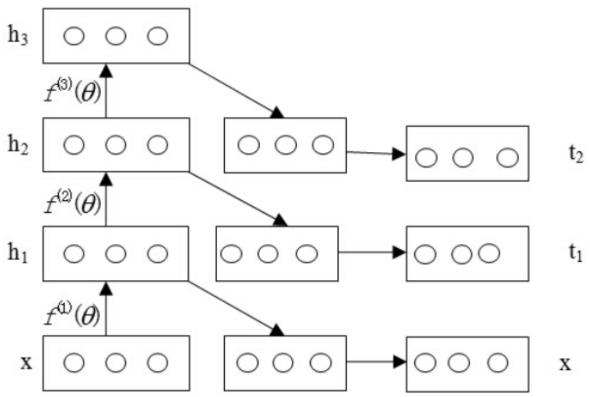

FIGURE 5. The architecture of the stacked auto-encoder with three hidden layers.

the features for the modalities of each object, we can get the feature vectors. For example, we can get three feature vectors, i.e., image feature vector, text feature vector and audio feature vector, for the object with three modalities including image, text and audio. In the second part, we use the vector outer product to fuse the learned feature vectors to obtain a feature tensor for each object. In the final part, we implement the tensor k-means algorithm based on the tensor distance that is used to measure the similarity between each object and every clustering center to cluster the heterogeneous objects represented by the feature tensors.

A. FEATURE LEARNING BASED ON DEEP LEARNING MODELS

To learn the features for each modality of every object, we build three stacked auto-encoders, each with three hidden layers. Fig. 5 shows the architecture of a stacked auto-encoder.

The stacked auto-encoder shown in Fig. 5 is built by stacking three basic auto-encoders from bottom to top. Assume that x denotes the input data, the basic auto-encoder projects x into the hidden layer h and the output layer y by two function, i.e., encoding function f and decoding function g, respectively:

Algorithm 1: Back-propagation Algorithm for Training Auto-encoder.

Input: $\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$, λ, $iter_{\max}$, th
Output: θ

for $iter = 1, \ldots, iter_{\max}$ **do**
  for $sample = 1, ..., m$ **do**
    for $j = 1, ..., p$ **do**
      $h_j^{(2)} = f(\sum_{i=1}^{n} W_{ji}^{(1)} + b_i^{(1)})$;
    for $i = 1, ..., n$ **do**
      $y_i^{(3)} = f(\sum_{j=1}^{p} W_{ij}^{(2)} a_j^{(2)} + b_i^{(2)})$;
    if $J(\theta) \geq th$ **then**
      for $i = 1, ..., n$ **do**
        $\delta_i^{(3)} = (y_i^{(3)} \cdot (1 - y_i^{(3)})) \cdot (y_i^{(3)} - x_i)$;
      for $j = 1, ..., p$ **do**
        $\delta_j^{(2)} = (\sum_{i=1}^{n} W_{ij}^{(2)} \delta_i^{(3)})(h^{(2)}(1 - h^{(2)}))$;
      for $i = 1, ..., n$ **do**
        $\Delta b_i^{(2)} = \Delta b_i^{(2)} + \delta_i^{(3)}$;
        for $j = 1, ..., p$ **do**
          $\Delta w_{ij}^{(2)} = \Delta w_{ij}^{(2)} + a_j^{(2)} \cdot \delta_i^{(3)}$;
      for $j = 1, ..., p$ **do**
        $\Delta b_j^{(1)} = \Delta b_j^{(1)} + \delta_j^{(2)}$;
        for $i = 1, ..., n$ **do**
          $\Delta w_{ji}^{(1)} = \Delta w_{ji}^{(1)} + x_i \cdot \delta_j^{(2)}$;
      $W = W - \eta \times (\frac{1}{N} \Delta w)$;
      $b = b - \eta \times (\frac{1}{N} \Delta b)$;

FIGURE 6. Algorithm 1.

$$h = f_\theta(W^{(1)}x + b^{(1)}). \quad (1)$$

$$y = s_\theta(W^{(2)}x + b^{(2)}). \quad (2)$$

In the projection functions, $\theta = (W^{(1)}, b^{(1)}; W^{(2)}, b^{(2)})$ represents the parameter, and each projection function usually uses the sigmoid function: $f(x) = 1/(1 + e^{-x})$.

The goal of the basic auto-encoder is to train the parameter $\theta = (W^{(1)}, b^{(1)}; W^{(2)}, b^{(2)})$ by using the following objective function:

$$J = [\frac{1}{m}\sum_{i=1}^{m}(\frac{1}{2}||y^{(i)} - x^{(i)}||^2)] + \lambda \sum_{ij} W_{ij}^2. \quad (3)$$

In the objective function, the first item is used to measure the error between the input and the output and the second item with a hyper-parameter λ aims to prevent over-fitting.

Perhaps the most well-known technique used to train the parameter $\theta = (W^{(1)}, b^{(1)}; W^{(2)}, b^{(2)})$ is the back-propagation algorithm by the following two steps, i.e., feed-forward step and back-propagation step, as illustrated in the first algorithm [26].

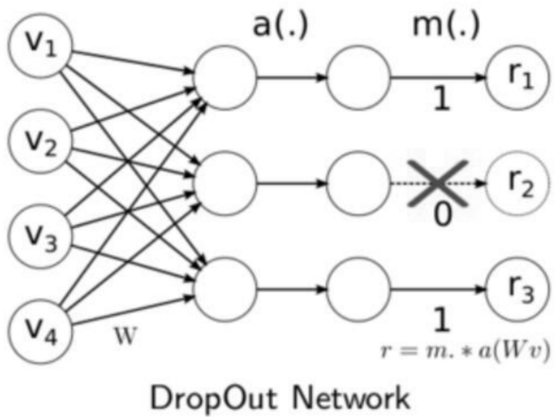

FIGURE 7. Example of the dropout auto-encoder.

Furthermore, to prevent the over-fitting, we adopt the dropout to get three dropout stacked auto-encoder models for feature learning on each object. In detail, we set each unit in every hidden layer of the stacked auto-encoder model to 0 with the possibility of 0.5. Fig. 7 illustrates an example of a dropout auto-encoder model.

B. FEATURE FUSION BASED ON VECTOR OUTER PRODUCT

In this section, we establish a future tensor for each object by using the vector outer product to fuse the learned feature vectors. Consider one multi-modal object x with three modalities, i.e., image modality, text modality and audio modality, we can get three feature vectors, i.e., image feature vector A, text feature vector B and audio feature vector C, for each object after feature learning in the last section. We can build a three-order feature tensor T, namely $T = A \circ B \circ C$, for the object x.

$\circ$ denotes the vector outer product, one of the commonly used algebraic operations [27]. Given a three-dimension vector $A = [a_1, a_2, a_3]$ and a two-dimension vector $B = [b_1, b_2]$, their outer product will produce a 2×3 matrix $T = A \circ B$, as shown as follows:

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \otimes \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} = \begin{bmatrix} a_1 b_1 & a_2 b_1 & a_3 b_1 \\ a_1 b_2 & a_2 b_2 & a_3 b_2 \end{bmatrix} \quad (4)$$

More generally, the outer product of n vectors will produce a n-order tensor.

C. TENSOR K-MEANS ALGORITHM BASED ON TENSOR DISTANCE

The conventional k-means algorithm could cluster the objects represented by vectors. However, after feature fusion, one feature tensor is built for each heterogeneous object. Therefore, the k-means algorithm could not cluster the heterogeneous objects represented by the feature tensors. To address this challenge, we devise a tensor k-means algorithm by using the tensor distance to compute the distance between the object and each clustering center.

To compute the tensor distance between two N-order tensors, i.e., $X, Y \in R^{I_1 \times I_2 \times \cdots \times I_N}$, we unfold each tensor X into the vector form x. Specially, $X_{i_1 i_2 \ldots i_N}$ is unfolded to x_l by $l = i_1 + \sum_{j=2}^{N} \prod_{t=1}^{j-1} I_t$. Therefore, the tensor distance d_{TD} between X and Y is computed by

$$d_{TD} = \sqrt{(x-y)^T G (x-y)}, \quad (5)$$

where x and y denote the vector form of the tensor X and the tensor Y, representatively, and G denotes the coefficient matrix used to reveal the location correlation between X and Y in the tensor space.

Therefore, the tensor k-means algorithm based on the tensor distance is outlined in the following four steps.

Step 1: Randomly select k objects as the clustering centers.

Step 2: Use the equation (5) to compute the tensor distance between each object and every clustering center and assign each object to the nearest clustering center.

Step 3: Recompute each clustering center.

Step 4: If convergence, stop the algorithm, otherwise, repeat Step 2 and Step 3.

From the steps of the tensor k-means algorithm, the key step is to compute the tensor distance between each object and every clustering center. Therefore, the tensor k-means algorithm has a computational complexity of $O(tnk)$, where t, n, k denote the number of iterations, objects and clustering centers, respectively.

IV. PERFORMANCE EVALUATION

In this section, we validate the high-order k-means (HOK-means) based on dropout deep learning models on two commonly used heterogeneous datasets collected from cyber-physical-social systems, i.e., NUS-WIDE and CUAVE. We compare the HOK-means algorithm with the HOPCM algorithm in two metrics, namely $E*$ and RI.

$E*$ is used to measure the accuracy of the produced clustering centers by computing the distance between the true clustering centers and the produced clustering centers in the following:

$$E* = \sqrt{\sum_{i=1}^{c} ||v_{ture}^i - v_*^i||^2}, \quad (6)$$

where v_{ture}^i and v_*^i represent the i-th true clustering center and the i-th produced clustering center by the approach $*$, c denotes the number of clustering centers.

RI aims to measure the clustering accuracy according to how many objects that are assigned into the correct groups. For example, if there are 1000 objects in the dataset and there are 950 objects are assigned into the correct groups, RI equals 95%.

Obviously, the lower $E*$ and the higher RI imply that the approach achieves the more accurate clustering result.

A. EXPERIMENTS ON NUS-WIDE

There are 269,648 images, each with some annotations, in the NUS-WIDE dataset that is collected from Flickr, a well-known image website. Zhang selected 80,000 representative images that can be clustered into 14 groups to evaluate the performance of the HOPCM algorithm. In this paper, we also uses the same images to compare the performance between the HOK-means algorithm and the HOPCM algorithm.

We perform the HOK-means algorithm and the HOPCM algorithm for five times and the experimental results are displayed in Fig. 8 and Fig. 9.

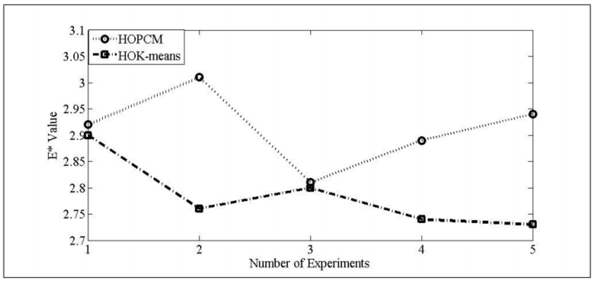

FIGURE 8. Experimental result on NUS-WIDE in $E*$.

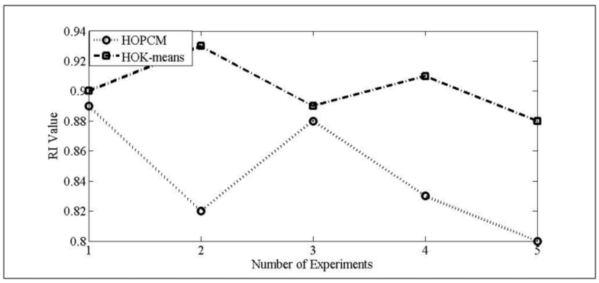

FIGURE 9. Experimental result on NUS-WIDE in RI.

The experimental results in $E*$ are displayed in Fig. 8. It can be seen from the results that the HOK-means algorithm achieves the lower $E*$ values than the HOPCM algorithm in five experiments. For example, in the second experiment, the HOK-means algorithm produces the $E*$ value with 2.76 while the HOPCM algorithm produces the $E*$ value with 3.01. The $E*$ value produced by the HOK-means algorithm is significantly smaller than that produced by the HOPCM algorithm. Such results demonstrate that the HOK-means algorithm produced more accurate clustering centers than the HOPCM algorithm.

Fig. 9 displays the experimental result in RI. We can see from Fig. 8 that HOK-means gets a bigger RI value than HOPCM in each experiment. Specially, the HOK-means algorithm produces the average RI value with 0.902 while the average RI value produced by the HOPCM is 0.844. Obviously, the HOK-means algorithm achieves the higher clustering accuracy than the HOPCM algorithm in terms of RI on the NUS-WIDE dataset.

B. EXPERIMENTS ON CUAVE

There are 1800 multi-modal objects in the CUAVE dataset that is widely used to validate the performance of the deep learning models and other clustering algorithms. This paper

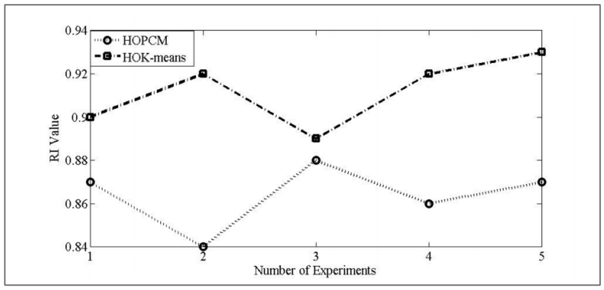

FIGURE 10. Experimental result on CUAVE in RI.

TABLE 1. Average execution time (minutes).

Dataset	HOPCM	HOK-means
NUS-WIDE	295	261
CUAVE	127	111

uses this dataset to estimate the HOK-means approach by comparison with the HOPCM approach.

Since this dataset does not have the true clustering centers, we compare the HOK-means algorithm with the HOPCM algorithm in the clustering accuracy with the metric RI. Each approach is performed on the CUAVE dataset for five times. Fig. 10 shows the experimental results.

We can make the several remarkable observations from the experimental results presented in Fig. 10. First, the HOK-means algorithm obtains bigger RI values than the HOPCM algorithm in five experiments. For instance, the HOK-means algorithm yields the RI value with 0.92 while the HOPCM algorithm yields the RI value with 0.86 in the fourth experiment. Second, the HOPCM algorithm produces the highest RI value with 0.88 in the second experiment. However, this value is still smaller than the RI value with 0.89 produced by the HOK-means algorithm in the second experiment. Finally, HOK-means and HOPCM yield the average RI values with 91.2% and 86.4%, respectively. Such observations clearly validate the better performance of the HOK-means algorithm compared with the HOPCM algorithm in terms of RI.

C. EXECUTION TIME

Finally, we compare the clustering efficiency between HOK-means and HOPCM with respect to average execution time for two datasets. Table 1 illustrates the results.

From Table 1, we can see that HOPCM is more time-consuming than HOK-means. Such results imply that our proposed approach is more efficient than HOPCM on two datasets. Therefore, our approach is potential to cluster large-scale data in cyber-physical-social systems. However, HOPCM is slightly more time-consuming than HOK-means since they have the same computational complexity.

V. CONCLUSION

In this paper, we proposed a high-order k-means algorithm based on dropout deep learning models for heterogeneous data clustering in cyber-physical-social systems. One remarkable point of the proposed method is to build different dropout stacked auto-encoder models, each with three hidden layers, to learn features for each modality of every object. Different from the conventional clustering approaches which use the vector to represent each object, we establish one feature tensor by using the vector outer product for every object in the heterogeneous dataset. To cluster the heterogeneous objects represented by the feature tensor, we devise a tensor k-means algorithm by using the tensor distance to measure the similarity between each object and the clustering centers. Experimental results implied that the proposed approach performed better than the HOPCM algorithm in the clustering accuracy.

In the future work, we will build the different deep learning models to learn features for different modalities in the clustering process. For example, the convolutional neural networks has achieved the best performance for large-scale images feature learning while the stacked auto-encoders performs best for text and audio feature learning, so we plan to build a convolutional neural network and the stacked auto-encoders to learn the features for the image modalities and other two modalities, representatively.

REFERENCES

[1] A. Sheth, P. Anantharam, and C. Henson, "Physical-cyber-social computing: An early 21st century approach," *IEEE Intell. Syst.*, vol. 28, no. 1, pp. 78–82, Jan. 2013.

[2] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "High-order possibilistic c-means algorithms based on tensor decompositions for big data in IoT," *Inf. Fusion*, vol. 39, pp. 72–80, Jan. 2018.

[3] A. Kamilaris and A. Pitsillides, "Mobile phone computing and the Internet of Things: A survey," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 885–898, Dec. 2016.

[4] Q. Zhang, C. Zhu, L. T. Yang, Z. Chen, L. Zhao, and P. Li, "An incremental CFS algorithm for clustering large data in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1193–1201, Jun. 2017.

[5] X. Yang, T. Zhang, and C. Xu, "Cross-domain feature learning in multimedia," *IEEE Trans. Multimedia*, vol. 17, no. 1, pp. 64–78, Jan. 2015.

[6] Q. Zhang and Z. Chen, "A distributed weighted possibilistic c-means algorithm for clustering incomplete big sensor data," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 5, p. 430814, 2014.

[7] T. C. Havens, J. C. Bezdek, C. Leckie, L. O. Hall, and M. Palaniswami, "Fuzzy c-means algorithms for very large data," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 6, pp. 1130–1146, Dec. 2012.

[8] L. Meng, A.-H. Tan, and D. Xu, "Semi-supervised heterogeneous fusion for multimedia data co-clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2293–2306, Sep. 2014.

[9] X. Cai, F. Nie, H. Huang, and F. Kamangar, "Heterogeneous image feature integration via multi-modal spectral clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 20–25.

[10] B. Gao, T.-Y. Liu, X. Zheng, Q.-S. Cheng, and W.-Y. Ma, "Consistent bipartite graph co-partitioning for star-structured high-order heterogeneous data co-clustering," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2005, pp. 41–50.

[11] B. Long, Z. Zhang, X. Wú, and P. S. Yu, "Spectral clustering for multi-type relational data," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 585–592.

[12] M. Rege, M. Dong, and J. Hua, "Graph theoretical framework for simultaneously integrating visual and textual features for efficient Web image clustering," in *Proc. Int. Conf. World Wide Web*, 2008, pp. 317–326.

[13] Y. Chen, L. Wang, and M. Dong, "Non-negative matrix factorization for semisupervised heterogeneous data coclustering," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1459–1474, Oct. 2010.

[14] Q. Zhang, L. T. Yang, Z. Chen, P. Li, and M. J. Deen, "Privacy-preserving double-projection deep computation model with crowdsourcing on cloud for big data feature learning," *IEEE Internet Things J.*, to be published. [Online]. Available: http://ieeexplore.ieee.org/document/7994599/, doi: 10.1109/JIOT.2017.2732735.

[15] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "PPHOPCM: Privacy-preserving high-order possibilistic c-means algorithm for big data clustering with cloud computing," *IEEE Trans. Big Data*, to be published. [Online]. Available: http://ieeexplore.ieee.org/document/7920374/, doi: 10.1109/TBDATA.2017.2701816.

[16] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010.

[17] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "An improved deep computation model based on canonical polyadic decomposition," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. [Online]. Available: http://ieeexplore.ieee.org/document/8015194/, doi: 10.1109/TSMC.2017.2701797.

[18] Y. Qi, Y. Wang, X. Zheng, and Z. Wu, "Robust feature learning by stacked autoencoder with maximum correntropy criterion," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2014, pp. 6716–6720.

[19] Q. Zhang, L. T. Yang, X. Liu, Z. Chen, and P. Li, "A tucker deep computation model for mobile multimedia feature learning," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 13, no. 3s, pp. 39:1–39:18, 2017.

[20] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[21] Y. Liu, Y. Liu, and K. C. C. Chan, "Tensor distance based multilinear locality-preserved maximum information embedding," *IEEE Trans. Neural Netw.*, vol. 21, no. 11, pp. 1848–1854, Nov. 2010.

[22] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "NUSWIDE: A real-world Web image database from national University of Singapore," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2009, Art. no. 48.

[23] E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. N. Gowdy, "CUAVE: A new audio-visual database for multimodal human-computer interface research," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2002, pp. 2017–2020.

[24] X.-X. Chen and X. Lin, "Big data deep learning: Challenges and perspectives," *IEEE Access*, vol. 2, pp. 514–525, 2014.

[25] M. A. Ranzato, Y.-L. Boureau and Y. LeCun, "Sparse feature learning for deep belief networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1185–1192.

[26] Q. Zhang, M. Lin, L. T. Yang, Z. Chen, and P. Li, "Energy-efficient scheduling for real-time systems based on deep Q-learning model," *IEEE Trans. Sustain. Comput.*, to be published. [Online]. Available: http://ieeexplore.ieee.org/document/8016380/, doi: 10.1109/TSUSC.2017.2743704.

[27] A. Cichocki. (Mar. 2014). "Era of big data processing: A new approach via tensor networks and tensor decompositions." [Online]. Available: https://arxiv.org/abs/1403.2048

FANYU BU received the B.S. degree from Inner Mongolia Agricultural University and the master's degree from Inner Mongolia University. He is currently a Lecturer with the Department of Biomedical Informatics, Inner Mongolia University of Finance and Economics, China. His research interests include cyber-physical-social systems and big data.