Research Article

A Deep Fusion Gaussian Mixture Model for Multiview Land Data Clustering

Peng Li,[1] **Zhikui Chen** [iD]**,**[1,2] **Jing Gao,**[1,2] **Jianing Zhang** [iD]**,**[1] **Shan Jin** [iD]**,**[1] **Wenhan Zhao** [iD]**,**[1] **Feng Xia,**[1,2] **and Lu Wang**[3,4,5]

[1]*School of Software Technology, Dalian University of Technology, Dalian 116620, China*
[2]*Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian 116620, China*
[3]*College of Natural Resources and Environment, South China Agricultural University, Guangzhou 510642, China*
[4]*Guangdong Province Key Laboratory for Land Use and Consolidation, South China Agricultural University, Guangzhou 510642, China*
[5]*Guangdong Province Engineering Research Center for Land Information Technology, South China Agricultural University, Guangzhou 510642, China*

Correspondence should be addressed to Zhikui Chen; zkchen@dlut.edu.cn

With the rapid industrialization and urbanization, pattern mining of soil contamination of heavy metals is attracting increasing attention to control soil contamination. However, the correlation over various heavy metals and the high-dimension representation of heavy metal data pose vast challenges on the accurate mining of patterns over heavy metals of soil contamination. To solve those challenges, a multiview Gaussian mixture model is proposed in this paper, to naturally capture complicated relationships over multiviews on the basis of deep fusion features of data. Specifically, a deep fusion feature architecture containing modality-specific and modality-common stacked autoencoders is designed to distill fusion representations from the information of all views. Then, the Gaussian mixture model is extended on the fusion representations to naturally recognize the accurate patterns of the intra- and inter-views. Finally, extensive experiments are conducted on the representative datasets to evaluate the performance of the multiview Gaussian mixture model. Results show the outperformance of the proposed methods.

1. Introduction

With the rapid industrialization and urbanization over the world, environmental contamination is attracting increasing attention nowadays, which is caused by unreasonable usage of natural resources, such as the overuse of coal [1]. Among the environmental contamination, the status of soil contamination of heavy metals is the core concern of the public, with the scare of the heavy metal security of agricultural products that easily have a direct influence on our health [2]. A large number of researchers force on the control of soil contamination of heavy metals by mining intrinsic patterns hidden over various heavy metals which can do a favor to the contamination control and environmental protection. However, the correlation over various heavy metals and the high-dimension representation of heavy metal data pose vast challenges on the accurate mining of patterns over heavy metals of soil contamination. With the continuous development of industrialization and urbanization, more research is still required to capture effective patterns of high-dimension representation of heavy metal data, to control the soil contamination.

In recent years, large amounts of research have been proposed to learn patterns of data to improve our lives [3–8]. For example, Chen et al. used multivariate statistics and geostatistics to explore distributions of heavy metals in the soil of northwest China, which can capture pollution sources of heavy metals based on patterns of distributions [9].

Additionally, the chemical mass balance model, factor analysis, target transformation factor analysis, and principal component analysis are used to capture the complicated relationship of heavy metals [10–12]. Those statistical methods are able to mine patterns of heavy metals in simple cases where there are not many kinds of heavy metals. Also, they can only mine contamination patterns in a single view. In other words, those traditional statistical methods cannot well learn complex contamination patterns of heavy metals in the current soil, which are expressed by high-dimension data. Thus, to explore the complicated patterns of various heavy metals requires novel computing methods.

Clustering, as a fundamental approach to pattern mining, divides data into several groups based on data similarity; hence, data in the same group are more similar than data in different groups [13]. It is widely used in various domains, such as text recognition and image processing [14–17]. Among various clustering algorithms, the Gaussian mixture model, as a generating method, captures each cluster by a probability distribution, which well fit multiview characteristics of data in a natural manner [18]. Inspired by this, a Gaussian mixture model is introduced to mine the multiview heavy metal data. However, the current Gaussian mixture model-based methods neglect the multiview information of data, especially the deep intrinsic fusion features of all views.

To solve those challenges, in this paper, a multiview Gaussian mixture model is proposed to naturally capture complicated relationships over multiviews on the basis of deep fusion features of data, which can potentially mine robust patterns of heavy metals in practice. In particular, a deep fusion feature architecture with modality-specific and modality-common stacked autoencoders is designed to distill fusion representations from the information of all views. Then, the Gaussian mixture model is extended on the fusion representations to naturally recognize the accurate patterns of the intra- and inter-views. Extensive experiments are conducted on the representative datasets to evaluate the performance of the multiview Gaussian mixture model. Results show that the proposed method can greatly outperform the compared methods.

Thus, the major contributions of this paper are threefold:

(i) To accurately capture complex patterns of heavy metal data, a multiview Gaussian mixture model is introduced based on the fusion representations, which fully considers information of each view in a nonlinear manner

(ii) To distill fusion representations from the information of all views, a deep fusion feature architecture is designed, which consists of modality-specific and modality-common stacked autoencoders

(iii) Extensive experiments with outperforming results are conducted to assess the performance on the representative datasets

The rest of the paper is organized as follows. Section 2 reviews common methods in statistical learning about the pattern mining of heavy metals. Sections 3 and 4 are the fundamentals of the proposed method. Section 5 describes the details of the proposed method, and Section 6 validates the proposed method. Finally, Section 7 concludes this work.

2. Related Works

To trace the source of the soil heavy metal pollution, a lot of statistical methods were proposed. Most of all can be grouped into the following:

Linear regression. Because of its simplicity and efficiency, linear regression is a frequently used method [19]. It tries to find the best linear projection function through updating the parameters of the function using the least square method or the gradient descent method. For example, Tian et al. [20] improved the multiple linear regression (MLR) method to quantitatively estimate relationships between soil properties and sources of heavy metals. In MLR, heavy metal concentrations were regarded as dependent variables while the scores of soil properties and sources were independent variables. However, due to the influence of various complex factors, such as climate, parent material, topography, and human activities, the linear projection cannot well model correlations between the environmental parameters and the soil properties in the practice of soil pollution research [21].

Decision tree. Decision tree methods such as classification and regression tree (CART) and random forest (RF) use a tree structure for deciding classification results by judging from the root to leaves [22, 23]. For example, Qiu et al. [24] applied stepwise linear regression (SLR), CART, and RF to the prediction of the soil Cd's spatial distribution. In that article, RF was the best method for handling the nonlinear and hierarchical relationships between soil Cd and influence factors. Wang et al. [25] aimed to use RF and the stochastic gradient boosting (SGB) method for identifying and apportioning heavy metal pollution. Both RF and SGB showed that the biggest reason for the concentrations of Pb and Cd was anthropogenic sources.

Neural network. The neural network imitates the mechanism of human brains, recombining the information of input to extract some simple and fuzzy features, producing the corresponding impression and judgment. Furthermore, nonlinear activation functions of each layer, such as the sigmoid function and Rectified Linear Unit (ReLU) function, play a great role in the nonlinear fitting ability. One representative work is [26]. Specifically, neural networks with Monte Carlo simulations are combined to address the uncertainties from data quality and measurement errors in predicting the copper's phytoavailability in contaminated soils against the soil input parameters.

Principal component analysis (PCA). The principal component analysis uses the covariance matrix of data matrix for choosing principal components of data so that it can eliminate the less important properties for reducing the dimension of data and extracting hidden subsets to detect possible sources. For surveying the Chinese farmland soil metal accumulation at the national scale, Niu et al. [27] performed multivariate statistical analysis on soil properties and metal concentrations using PCA and correlation analysis. Research results on 11 metals showed that Pb, Cd, Zn, and

Cu had the concentrations above reference values. At the same time, results indicated that the 4 metals' accumulation may be associated with artificial fertilization. Also, Sun et al. [28] used PCA and correlation coefficient analysis to mine the agricultural soil major and trace element accumulation in the Gannan area, China. More PCA-based research includes [29, 30].

Cluster analysis (CA). CA classifies the data points into several disjoint and nonempty clusters on the basis of the similarity or distance among data points. There are various clustering algorithms used in the heavy metal analysis, such as spectral clustering, K-means, and hierarchical clustering. For the characterization of heavy metals in soils, Chai et al. [31] performed PCA and clustering analysis on data from the surface and underlying horizons of grassland. Three principal components were extracted, and hierarchical clustering proved this result. Moreover, in the three clusters from hierarchical clustering, clusters 1 and 2 were merged at a higher level so that the heavy metals in clusters 1 and 2 had a similar source. Similarly, Liu et al. [32] applied PCA and clustering analysis on data from the outskirts of Changchun, China. Results showed that Pb, Cu, and Zn were from human activities, while Cr and Ni were from natural sources.

In summary, the above methods can mine patterns of heavy metals in soil in simple cases where there are not many kinds of heavy metals. However, they neglect the multiview characteristics of land data, leading to undesired result patterns in complicated cases. Also, those methods cannot capture intrinsic patterns within high-dimension representations of land data. To solve those challenges, a deep fusion Gaussian mixture model for multiview land data clustering is proposed in this paper.

3. The Deep Stacked Autoencoder

The deep stacked autoencoder is a neural network of the fully connected paradigm on the basis of autoencoders, as shown in Figure 1 [33–35]. It extracts instinct representations of data by data reconstruction between an encoder and a decoder where the encoder constructs deeper representations layer by layer with the decoder reconstructing the input [36–38]. The deep stacked autoencoder is trained by a greedy layer-wise method in which each layer in the encoder and the corresponding layer of the decoder are modeled as an autoencoder to obtain the pretrained parameters followed by an end-to-end fine-tuning training.

Specifically, in a deep stacked autocoder of l layers, the s-th layer is modeled as an autoencoder with the $(l - s + 1)$-th layer to pretrain weights and biases in the following form:

$$
\begin{aligned}
h^s &= f\left(w^s \odot h^{s-1} + b^s\right), \\
h^{l-s+1} &= f\left(w^{l-s+1} \odot h^s + b^{l-s+1}\right),
\end{aligned}
\tag{1}
$$

where w^s, w^{l-s+1}, b^s, and b^{l-s+1} are the weights and biases of the s-th layer and the $(l - s + 1)$-th layer, respectively. $\odot$ is the matrix product. h denotes the hidden representation.

After the pretraining, each hidden layer in the deep stacked autocoder is fine-tuned as follows:

$$
h^s = f\left(w^s \odot h^{s-1} + b^s\right),
\tag{2}
$$

which is based on the stochastic gradient descent algorithm.

4. The Gaussian Mixture Model

A Gaussian mixture model (GMM) is a generative probabilistic model with trainable parameters [16]. It uses several basis Gaussian components to naturally represent multimodal characteristics of collected data by a weighted superposition operation, where each Gaussian component denotes a modal source. Generally, the Gaussian mixture model is trained by the expectation-maximization method by maximizing the likelihood function, where the expectation step computes probability distributions of each sample generated from each basis component and the maximization step learns the mean, covariance, and weight parameters of each basis component. GMMs have been widely used in various applications, such as text clustering and image recognition.

Given a dataset $X = \{x_1, x_2, \cdots, x_N\}$ with $x_i \in R^d$, the Gaussian mixture distributions are denoted as

$$
p(x_i) = \sum_{k=1}^{K} w_k g(x_i ; \mu_k, \Sigma_k),
\tag{3}
$$

where w_k is the weight of each basis Gaussian component and $g(x_i ; \mu_k, \Sigma_k)$ represents the basis distribution parameterized by the mean vector μ_k and the covariance matrix Σ_k with the following form:

$$
g(x_i ; \mu_k, \Sigma_k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} \exp\left\{-\frac{1}{2}(x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k)\right\}.
\tag{4}
$$

The d is the dimension of data, and K is the number of basis Gaussian components.

Thus, to fit the given dataset $X = \{x_1, x_2, \cdots, x_N\}$, the logarithm likelihood function of GMM is expressed in the following form:

$$
\begin{aligned}
\log L &= \log \left(\prod_{i=1}^{N} p(x_i, z_i)\right) \\
&= \log \left(\prod_{i=1}^{N} \prod_{k=1}^{K} (w_k g(x_i ; \mu_k, \Sigma_k))^{z_{ik}}\right) \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} (z_{ik} \log(w_k) + z_{ik} \log(g(x_i ; \mu_k, \Sigma_k))),
\end{aligned}
\tag{5}
$$

where $z_i \in \{0, 1\}^K$, $\sum_{k=0}^{K} z_{ik} = 1$, denotes the component from which x_i is generated. Then, setting the derivates of $\log L$ to
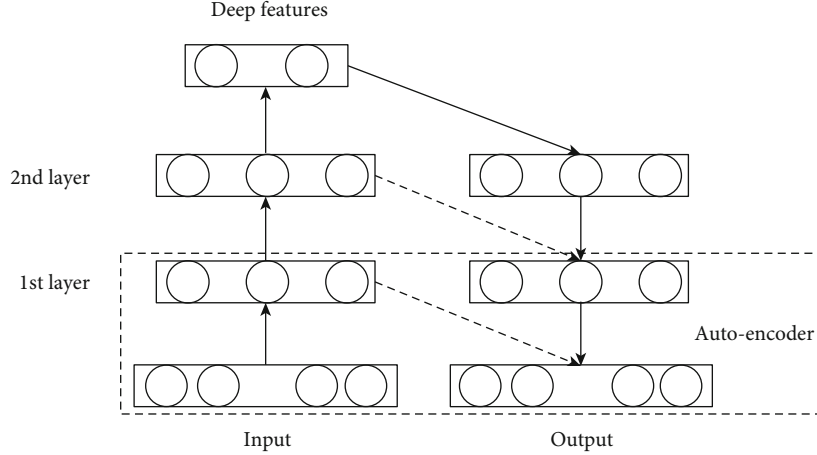
Deep features

FIGURE 1: The computing paradigm of the stacked autoencoder.
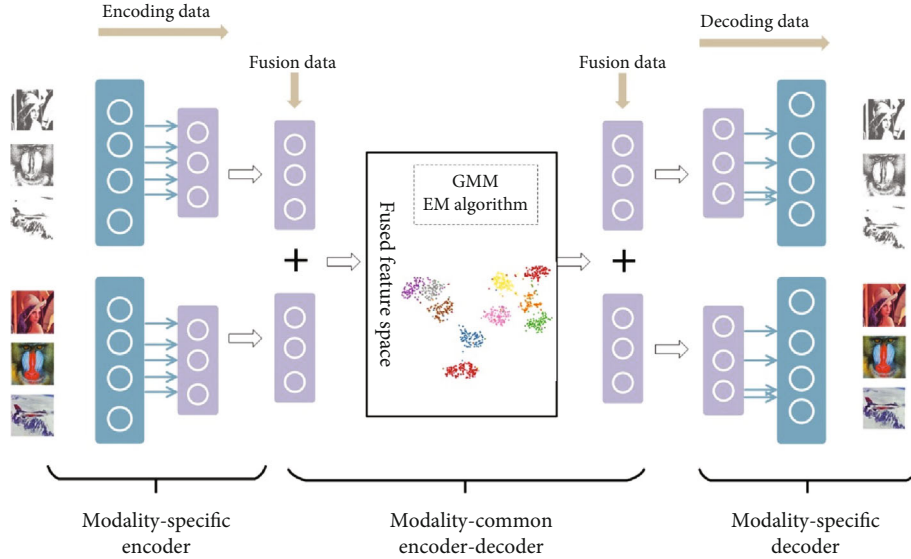

FIGURE 2: The computing paradigm of the multiview fusion Gaussian mixture model. Modality-specific encoders, modality-common encoder-decoder, and modality-specific decoders are linked in a cascaded manner where data are transferred into hidden representations of each view by modality-specific encoders; then, those hidden representations are concentrated, which are reconstructed via the modality-common encoder-decoder, and finally, the reconstructed hidden representations are decoded into the original data space by modality-specific decoders.

be zero, we can get the computing equations of the mean, covariance, and weight parameters of each basis component.

$$w_k = \frac{1}{N} \sum_{i=1}^{N} r_{ik},$$

$$\mu_k = \frac{\sum_{i=1}^{N} r_{ik} x_i}{\sum_{i=1}^{N} r_{ik}}, \qquad (6)$$

$$\Sigma_k = \frac{\sum_{i=1}^{N} r_{ik} (x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{N} r_{ik}},$$

in which

$$r_{ik} = p(z_{ik} = 1 \mid x_i ; w_k, \mu_k, \Sigma_k) = \frac{w_k g(x_i ; \mu_k, \Sigma_k)}{\sum_{k=1}^{K} w_k g(x_i ; \mu_k, \Sigma_k)}. \qquad (7)$$

Generally, the expectation-maximization method is used to train GMM in an iterative maximization manner where current parameters are employed to estimate future parameters.

5. The Multiview Fusion Gaussian Mixture Model Algorithm

To mine complicated fusion relationships over multiview data, a deep fusion representation-based Gaussian mixture model is proposed, which is composed of the deep fusion feature learning and the expectation-maximization clustering. In the deep fusion feature learning, intrinsic view-specific features are first extracted by each view-specific stacked autoencoder. Then, those view-specific features are concentrated via a view-common stacked autoencoder, capturing fusion representations of multiview data. In the expectation-

maximization clustering, the Gaussian mixture model is used to recognize structure patterns of complicated shapes.

5.1. The Deep Fusion Feature Learning. To obtain the effective representations of multiview data, a deep fusion architecture is designed on the basis of the unsupervised encode-decode manner, which can avoid the dimensionality curse of data. As shown in Figure 2, in the deep fusion architecture, all the views of data are simultaneously fed into the corresponding view-specific stacked autoencoders, learning intrinsic view-specific features.

In detail, given the multiview dataset $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ in which each sample $\mathbf{x}_i$ is composed of v views $\mathbf{x}_i = (x_i^1, x_i^2, \cdots, x_i^v)$, each sample $\mathbf{x}_i$ is mapped to the view-specific feature space as follows:

$$
\begin{aligned}
h_i^1 &= f_l^1\left(f_{l-1}^1\left(\cdots\left(f_1^1\left(x_i^1\right)\right)\right)\right), \\
&\vdots \\
h_i^j &= f_l^j\left(f_{l-1}^j\left(\cdots\left(f_1^j\left(x_i^j\right)\right)\right)\right), \\
&\vdots \\
h_i^v &= f_l^v\left(f_{l-1}^v\left(\cdots\left(f_1^v\left(x_i^v\right)\right)\right)\right),
\end{aligned}
\tag{8}
$$

where h_i^j is the feature of the j-th view and $f_l^j(f_{l-1}^j(\cdots(f_1^j())))$ is the corresponding encoding network function with the trainable parameters $w_l^j, \cdots, w_1^j$ and $b_l^j, \cdots, b_1^j$. To train those parameters, the features of all views are mapped to original data space as follows:

$$
\begin{aligned}
x_i^1 &= g_l^1\left(g_{l-1}^1\left(\cdots\left(g_1^1\left(h_i^1\right)\right)\right)\right), \\
&\vdots \\
x_i^j &= g_l^j\left(g_{l-1}^j\left(\cdots\left(g_1^j\left(h_i^j\right)\right)\right)\right), \\
&\vdots \\
x_i^v &= g_l^v\left(g_{l-1}^v\left(\cdots\left(g_1^v\left(h_i^v\right)\right)\right)\right),
\end{aligned}
\tag{9}
$$

where $g_l^j(g_{l-1}^j(\cdots(g_1^j())))$ denotes the decoding network function. The view-specific encoder is cascaded by the corresponding decoder to get the pretrained weights and biases with the help of the stochastic gradient descent algorithm via the end-to-end training.

After the view-specific intrinsic representations $\{h_i^1, h_i^2, \cdots, h_i^v\}$ are obtained; they are concentrated in the following form:

$$
\mathbf{h}_i = \mathrm{con}\left(h_i^1, h_i^2, \cdots, h_i^v\right),
\tag{10}
$$

where $\mathrm{con}()$ is the linear concentration function. Then, a view-common stacked autoencoder is used to transfer the concentrated representations to a fusion feature space, learning fused representations of multiview data via

$$
\begin{aligned}
\mathbf{h}_i^{\mathrm{fusion}} &= \mathrm{encoder}(\mathbf{h}_i), \\
\mathbf{h}_i &= \mathrm{decoder}\left(\mathbf{h}_i^{\mathrm{fusion}}\right),
\end{aligned}
\tag{11}
$$

in which $\mathrm{encoder}()$ and $\mathrm{decoder}()$ are deep neural networks with the same number of layers.

5.2. The Clustering Pattern Mining. Specifically, after obtaining the fusion representations of the multiview dataset $\{f_1, f_2, \cdots, f_N\}$, the Gaussian mixture model with K basis components is defined as follows:

$$
p(f_i) = \sum_{k=1}^{K} w_k g(f_i; \mu_k, \Sigma_k),
\tag{12}
$$

where w_k denotes the weight of the k-th basis Gaussian model, f_i represents the i-th fusion representation, and $g(f_i; \mu_k, \Sigma_k)$ is the basis distribution parameterized by the mean vector μ_k and the covariance matrix Σ_k with the following form:

$$
g(f_i; \mu_k, \Sigma_k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} \exp\left\{-\frac{1}{2}(f_i - \mu_k)^T \Sigma_k^{-1}(f_i - \mu_k)\right\}.
\tag{13}
$$

The d is the dimension of fusion representations of data.

Thus, the logarithm likelihood function of the given data is expressed in the following form:

$$
\begin{aligned}
\log L &= \log\left(\prod_{i=1}^{N} p(f_i, z_i)\right) \\
&= \log\left(\prod_{i=1}^{N}\prod_{k=1}^{K}(w_k g(f_i; \mu_k, \Sigma_k))^{z_{ik}}\right) \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K}(z_{ik}\log(w_k) + z_{ik}\log(g(f_i; \mu_k, \Sigma_k))),
\end{aligned}
\tag{14}
$$

where $z_i \in \{0, 1\}^K$, $\sum_{k=0}^{K} z_{ik} = 1$, denotes the component from which f_i is generated.

Then, setting the derivates of $\log L$ to be zero, we can get the computing equations of the mean, covariance, and weight parameters of each basis component.

$$
\begin{aligned}
w_k &= \frac{1}{N}\sum_{i=1}^{N} r_{ik}, \\
\mu_k &= \frac{\sum_{i=1}^{N} r_{ik} f_i}{\sum_{i=1}^{N} r_{ik}}, \\
\Sigma_k &= \frac{\sum_{i=1}^{N} r_{ik}(f_i - \mu_k)(f_i - \mu_k)^T}{\sum_{i=1}^{N} r_{ik}},
\end{aligned}
\tag{15}
$$

The multiview fusion Gaussian mixture model algorithm.
Input: the multiview dataset $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$, the number of component models K, the hyperparameters of deep fusion architecture
Output: patterns of the input data
1. To randomly initialize parameters of each autoencoder in the deep fusion architecture;
2. To train each autoencoder layer by layer;
3. To fine-tune the deep fusion architecture in an end-to-end manner;
4. To randomly initialize model parameters and weight coefficients of Gaussian models;
5. To compute the probability of each sample generated from each Gaussian model;
6. To compute the model parameters and weight coefficients of each Gaussian model;
7. To update model parameters and weight coefficients of Gaussian models;
8. Go to 5 until convergence, then output the probability of each data sample generated from each Gaussian model as patterns of the input data.

ALGORITHM 1

TABLE 1: ARI results of models on the basis of raw representations.

Models	K-means-M	GMM-M
ARI	0.36	0.24

TABLE 2: NMI results of models on the basis of raw representations.

Models	K-means-M	GMM-M
NMI	0.49	0.37

in which

$$r_{ik} = p(z_{ik} = 1 \mid f_i; w_k, \mu_k, \Sigma_k) = \frac{w_k g(f_i; \mu_k, \Sigma_k)}{\sum_{k=1}^{K} w_k g(f_i; \mu_k, \Sigma_k)}. \quad (16)$$

5.3. The Multiview Fusion Gaussian Mixture Model Algorithm. The multiview fusion Gaussian mixture model algorithm consists of two steps, i.e., fusion feature learning and pattern mining. In the former step, all view-specific stacked autoencoders and view-common stacked autoencoders are trained in a greedy layer-wise unsupervised manner. Then, an end-to-end fine-tuning training is conducted on the basis of SGD. In the latter step, the fusion features of multiview data extracted in the former step are fed into the multiview Gaussian mixture model with the predefined K. Then, the parameters in each component Gaussian model and weight coefficients between Gaussian models are learned based on the expectation-maximization algorithm. The details of the multiview fusion Gaussian mixture model algorithm are shown in Algorithm 1.

6. Experiments

To evaluate the performance of the multiview fusion Gaussian mixture model, extensive experiments are conducted on two datasets. Those experiments are implemented by Python, and the details of the experiments are described in the following.

TABLE 3: ARI results of models on the basis of deep representations.

Models	K-means-DM	GMM-DM	K-means-DE	GMM-DE	Clustering-DF
ARI	0.65	0.76	0.57	0.74	0.80

TABLE 4: NMI results of models on the basis of deep representations.

Models	K-means-DM	GMM-DM	K-means-DE	GMM-DE	Clustering-DF
NMI	0.71	0.81	0.62	0.80	0.85

6.1. Compared Methods. K -means. K-means is a typical clustering method that is widely used in practice as a representative baseline.

Gaussian mixture model. The Gaussian mixture model is a generative method based on the probability distribution. It mines cluster patterns of data by multiple Gaussian distributions.

In the experiments, the K-means and Gaussian mixture model are used as the based model, which are extended to modality-specific, modality-common, modality-fused methods with respect to raw, shallow, and deep representations of data.

6.2. Datasets. MNIST-EMNIST. MNIST [39] and EMNIST [40] are the representative datasets of images, which contain images of numbers from 0 to 9. They are widely used in image classification and image clustering. In the experiments, MNIST and EMNIST are fed into a fully connected neural network and a convolutional neural network, respectively, in feature learning to represent different views. The results are illustrated in Tables 1–4. Also, Figure 3 visualizes the feature learning processing.

6.3. Results. In the results of Tables 1–4, K-means-M and GMM-M are the traditional K-means and GMM clustering algorithms conducted on the raw representations of MNIST. K-means-DM and GMM-DM denote the K-means and GMM performed on the deep representations of MNIST, which is extracted by the modality-specific stacked autoencoder. K-means-DE and GMM-DE are similar models
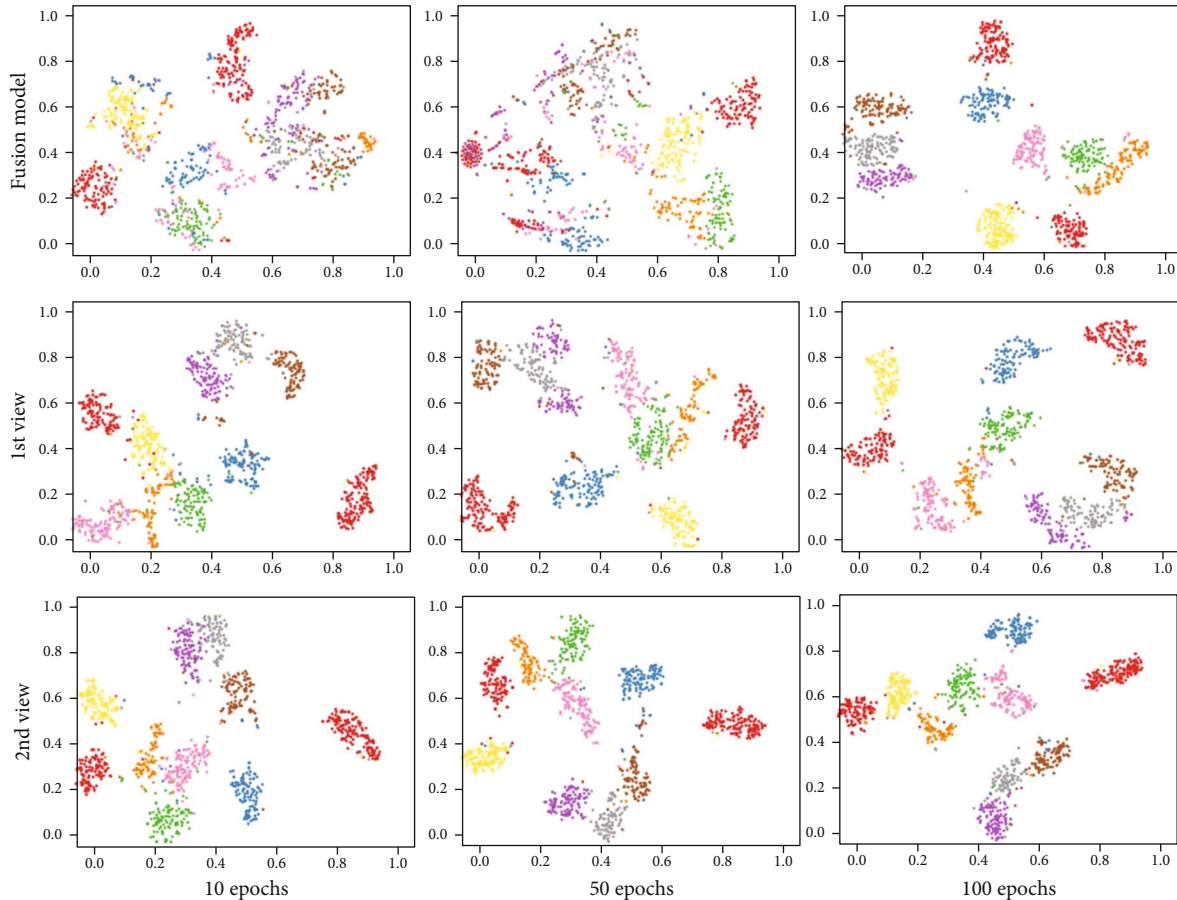
FIGURE 3: The t-SNE figures of each view model and the fusion model.

performed on EMNIST. Clustering-DF denotes the results of the proposed model.

From the above results, several observations can be concluded. In raw representations of data, K-means produced better results than GMM in terms of ARI and NMI. This is because the less important properties in raw representations are also modeled by the probability distributions of GMM, decreasing the clustering performance. The second observation is that the deep feature-based methods (K-means-DM, GMM-DM) outperform the shallow methods (K-means-M, GMM-M), since the proposed modality-specific stacked autoencoder can well extract intrinsic features of each view of data. Additionally, the clustering results of GMM-DM are better than those of K-means-DM, since the multiple Gaussian distributions in GMM can better fit patterns of data than the hard division in K-means with the clear features. The third observation is that the proposed method achieves the best results in terms of ARI and NMI, since it can distill information from all views by the designed deep fusion network. The observations of the results demonstrate the outperformance of the proposed method.

Figure 3 shows the t-SNE figures of the above models to visualize features learned by each model. There are two observations. First, the fusion model learns better representations than each single-view model. Specifically, the proposed model produces features where the distance of similar data is closer than that of dissimilar data, shown in the third column. Furthermore, the distance between different clusters is further. Second, the proposed model learns data representations faster than single-view models. In detail, the representations produced by the fusion model are more disorderly than those by the compared models at the beginning, while the fusion model achieves better representations after the same number of training epochs.

7. Conclusions

In this paper, a deep fusion Gaussian mixture model is proposed for multiview data clustering based on deep fusion representations, which can potentially capture intrinsic patterns of heavy metal data. In this model, a deep fusion feature architecture of modality-specific and modality-common stacked autoencoders is designed to merge fusion information of all views of data, which can well capture deep intrinsic fusion representations of data. Afterward, the Gaussian mixture model is extended on the fusion representations to naturally recognize the accurate patterns. Finally, results show the outperformance of the proposed methods by extensive experiments. In the future, more effective deep clustering methods will be explored, which are trained in an end-to-end manner.

Data Availability

The datasets used in this paper are public datasets which can be accessed by the following websites: MNIST and EMNIST (https://pytorch.org/docs/stable/torchvision/datasets.html).

Conflicts of Interest

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

Authors' Contributions

Peng Li, Jing Gao, Jianing Zhang, Shan Jin, and Wenhan Zhao are the first coauthors.

Acknowledgments

References

[1] X. Yang, L. Geng, and K. Zhou, "Environmental pollution, income growth, and subjective well-being: regional and individual evidence from China," *Environmental Science and Pollution Research*, vol. 27, no. 27, pp. 34211–34222, 2020.

[2] X. Zhao, Y. Sun, J. Huang, H. Wang, and D. Tang, "Effects of soil heavy metal pollution on microbial activities and community diversity in different land use types in mining areas," *Environmental Science and Pollution Research*, vol. 27, no. 16, pp. 20215–20226, 2020.

[3] R. Vamanan and K. Ramar, "Classification of agricultural land soils a data mining approach," *International Journal of Computer Science and Engineering*, vol. 3, no. 1, pp. 379–384, 2011.

[4] X. Wang, Z. Ning, S. Guo, and L. Wang, "Imitation learning enabled task scheduling for online vehicular edge computing," *IEEE Transactions on Mobile Computing*, p. 1, 2020.

[5] Z. Ning, K. Zhang, X. Wang, L. Guo, and R. Y. K. Kwok, "Intelligent edge computing in internet of vehicles: a joint computation offloading and caching solution," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–14, 2020.

[6] Z. Ning, P. Dong, X. Wang et al., "Mobile edge computing enabled 5G health monitoring for internet of medical things: a decentralized game theoretic approach," *IEEE Journal on Selected Areas in Communications*, vol. 2020, pp. 1–16, 2020.

[7] Z. Ning, P. Dong, X. Wang et al., "Partial computation offloading and adaptive task scheduling for 5G-enabled vehicular networks," *IEEE Transactions on Mobile Computing*, p. 1, 2020.

[8] X. Wang, Z. Ning, and S. Guo, "Multi-agent imitation learning for pervasive edge computing: a decentralized computation offloading algorithm," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 2, pp. 411–425, 2021.

[9] T. Chen, Q. Chang, J. Liu, J. G. P. W. Clevers, and L. Kooistra, "Identification of soil heavy metal sources and improvement in spatial mapping based on soil spectral information: a case study in northwest China," *Science of the Total Environment*, vol. 565, pp. 155–164, 2016.

[10] G. Shi, J. Liu, H. Wang et al., "Source apportionment for fine particulate matter in a Chinese city using an improved gas-constrained method and comparison with multiple receptor models," *Environmental Pollution*, vol. 233, pp. 1058–1067, 2018.

[11] S. Jain, S. K. Sharma, T. K. Mandal, and M. Saxena, "Source apportionment of PM10 in Delhi, India using PCA/APCs, UNMIX and PMF," *Particuology*, vol. 37, pp. 107–118, 2018.

[12] K. Keerthi, N. Selvaraju, and L. A. Varghese, "Use of combined receptor modeling technique for prediction of possible sources of particulate pollution in Kozhikode, India," *International Journal of Environmental Science and Technology*, vol. 17, no. 5, pp. 2623–2636, 2020.

[13] E. Min, X. Guo, Q. Liu, G. Zhang, J. Cui, and J. Long, "A survey of clustering with deep learning: from the perspective of network architecture," *IEEE Access*, vol. 6, pp. 39501–39514, 2018.

[14] M. Ibrar, J. Mi, S. Karim, A. A. Laghari, S. M. Shaikh, and V. Kumar, "Improvement of large-vehicle detection and monitoring on CPEC route," *3d Research*, vol. 9, no. 3, article 45, 2018.

[15] S. Karim, Y. Zhang, S. Yin, A. A. Laghari, and A. A. Brohi, "Impact of compressed and down-scaled training images on vehicle detection in remote sensing imagery," *Multimedia Tools and Applications*, vol. 78, no. 22, pp. 32565–32583, 2019.

[16] S. Karim, I. A. Halepoto, A. Manzoor, N. H. Phulpoto, and A. A. Laghari, "Vehicle detection in satellite imagery using maximally stable extremal regions," *International Journal of Computer Science and Network Security*, vol. 18, no. 4, 2018.

[17] A. A. Laghari, H. He, M. Shafiq, and A. Khan, "Assessment of quality of experience (QoE) of image compression in social cloud computing," *Multiagent and Grid Systems*, vol. 14, no. 2, pp. 125–143, 2018.

[18] C. E. Rasmussen, "The infinite Gaussian mixture model," *Advances in Neural Information Processing Systems*, vol. 12, pp. 554–560, 2000.

[19] J. A. Thompson, E. M. Pena-Yewtukhiw, and J. H. Grove, "Soil–landscape modeling across a physiographic region: topographic patterns and model transportability," *Geoderma*, vol. 133, no. 1-2, pp. 57–70, 2006.

[20] K. Tian, W. Hu, Z. Xing, B. Huang, M. Jia, and M. Wan, "Determination and evaluation of heavy metals in soils under two different greenhouse vegetable production systems in eastern China," *Chemosphere*, vol. 165, pp. 555–563, 2016.

[21] X. Zhang, F. Lin, Y. Jiang, K. Wang, and M. T. F. Wong, "Assessing soil Cu content and anthropogenic influences using decision tree analysis," *Environmental Pollution*, vol. 156, no. 3, pp. 1260–1267, 2008.

[22] G. De'ath and K. E. Fabricius, "Classification and regression trees: a powerful yet simple technique for ecological data analysis," *Ecology*, vol. 81, no. 11, pp. 3178–3192, 2000.

[23] J. M. Drake, C. Randin, and A. Guisan, "Modelling ecological niches with support vector machines," *Journal of Applied Ecology*, vol. 43, no. 3, pp. 424–432, 2006.

[24] L. Qiu, K. Wang, W. Long, K. Wang, W. Hu, and G. S. Amable, "A comparative assessment of the influences of human impacts on soil cd concentrations based on stepwise linear regression, classification and regression tree, and random forest models," *PLoS One*, vol. 11, no. 3, article e0151131, 2016.

[25] Q. Wang, Z. Xie, and F. Li, "Using ensemble models to identify and apportion heavy metal pollution sources in agricultural soils on a local scale," *Environmental Pollution*, vol. 206, pp. 227–235, 2015.

[26] N. Hattab, R. Hambli, M. Motelica-Heino, and M. Mench, "Neural network and Monte Carlo simulation approach to investigate variability of copper concentration in phytoremediated contaminated soils," *Journal of Environmental Management*, vol. 129, pp. 134–142, 2013.

[27] L. Niu, F. Yang, C. Xu, H. Yang, and W. Liu, "Status of metal accumulation in farmland soils across China: from distribution to risk assessment," *Environmental Pollution*, vol. 176, pp. 55–62, 2013.

[28] G. Sun, Y. Chen, X. Bi et al., "Geochemical assessment of agricultural soil: a case study in Songnen-Plain (Northeastern China)," *Catena*, vol. 111, pp. 56–63, 2013.

[29] Y. Shan, M. Tysklind, F. Hao, W. Ouyang, S. Chen, and C. Lin, "Identification of sources of heavy metals in agricultural soils using multivariate analysis and GIS," *Journal of Soils and Sediments*, vol. 13, no. 4, pp. 720–729, 2013.

[30] Y. Li, H. Gao, L. Mo, Y. Kong, and I. Lou, "Quantitative assessment and source apportionment of metal pollution in soil along Chao River," *Desalination and Water Treatment*, vol. 51, no. 19-21, pp. 4010–4018, 2013.

[31] Y. Chai, J. Guo, S. Chai, J. Cai, L. Xue, and Q. Zhang, "Source identification of eight heavy metals in grassland soils by multivariate analysis from the Baicheng–Songyuan area, Jilin Province, Northeast China," *Chemosphere*, vol. 134, pp. 67–75, 2015.

[32] L. Qiang, L. Jingshuang, W. Qicun, and W. Yang, "Source identification and availability of heavy metals in peri-urban vegetable soils: a case study from China," *Human and Ecological Risk Assessment*, vol. 22, no. 1, pp. 1–14, 2016.

[33] J. Gao, P. Li, Z. Chen, and J. Zhang, "A survey on deep learning for multimodal data fusion," *Neural Computation*, vol. 32, no. 1, pp. 1–36, 2020.

[34] J. Gao, P. Li, and Z. Chen, "A canonical polyadic deep convolutional computation model for big data feature learning in internet of things," *Future Generation Computer Systems*, vol. 99, pp. 508–516, 2019.

[35] P. Li, Z. Chen, L. T. Yang, Q. Zhang, and M. J. Deen, "Deep convolutional computation model for feature learning on big data in internet of things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 790–798, 2018.

[36] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," *Information Fusion*, vol. 42, pp. 146–157, 2018.

[37] P. Li, Z. Chen, L. T. Yang, J. Gao, Q. Zhang, and M. J. Deen, "An improved stacked auto-encoder for network traffic flow classification," *IEEE Network*, vol. 32, no. 6, pp. 22–27, 2018.

[38] Z. Ning, R. Y. K. Kwok, K. Zhang et al., "Joint computing and caching in 5G-envisioned internet of vehicles: a deep reinforcement learning based traffic control system," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12, 2020.

[39] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[40] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "EMNIST: an extension of MNIST to handwritten letters," 2017, https://arxiv.org/abs/1702.05373.