Deep Constrained Clustering with Active Learning

Dan HUANG*, Ran WEN, Boren DING, Junhua LI

Institute of Intelligence Technology, Geely University of China, Chengdu, 641423, China
danhuang@my.swjtu.edu.cn (**Corresponding author*), dbr2001916@163.com, junhua802@163.com

Abstract: Deep semi-supervised clustering approaches, which use supervised data to help the deep neural network acquire cluster-friendly representations, have improved clustering performance and simultaneously increased the semantic value of the clustering results. However, the majority of them cannot utilize both labeled and unlabeled data completely. Furthermore, in these methods, the supervised information is either passively acquired or randomly picked, which may be insufficient, redundant, and even decrease the performance of these models. This paper provides a deep semi-supervised clustering technique with active learning to address the problems mentioned above. The procedure is divided into two sections: model training and data labeling. In the model training section, the paired data is used to train the pseudo-Siamese network, and then the sub networks of the pseudo-Siamese network are fine-tuned using self-training. A new query strategy is devised in the data labeling part, which combines the traditional uncertainty query strategy with the deep Bayesian uncertainty query strategy. Finally, substantial tests are conducted to confirm the utility of the suggested approach on certain real-world data sets. The results of the tests demonstrate that both the suggested method and query strategy are practical.

Keywords: Semi-supervised clustering, Deep learning, Pairwise constraints, Active learning, Pseudo-Siamese network.

1. Introduction

Cluster analysis (Kwon & Lee, 2023; Hussein et al., 2021) is an unsupervised method that separates the data into various clusters according to the internal structural characteristics of the data set. Consequently, there is a high degree of similarity between data from the same cluster while there is a low degree of similarity between data from different clusters. However, data can be clustered at various levels to produce different clustering results, the possible outcomes without supervision information are unsuitable for downstream tasks (Nie et al., 2020).

Some researchers proposed a semi-supervised clustering technique (Safari & Afsari, 2020; Jia et al., 2020) which uses supervised data to drive clustering allocation. This method has a higher interpretability and has significantly improved clustering performance. However, with the popularization of the Internet and the development of computer technology, the data scale increases, and the data becomes more sophisticated and high-dimensional (Kim, 2019). On this type of data, most semi-supervised clustering algorithms perform poorly.

Deep learning was proposed by Hinton & Salakhutdinov (2006), which is a valuable feature learning method that employs the concept of hierarchical abstraction to abstract higher-level concepts into low-level feature space. So, scholars have introduced deep learning to learn cluster-friendly representation for semi-supervised clustering to address the above issues. For example, Ren et al. (2019) added the penalty term of pairwise constraint in deep embedded clustering (DEC) (Xie et al., 2016), which makes the connection point closer and the disconnection point farther away in the learned embedding space. Ohi et al. (2020) used the pre-trained DNN from ImageNet to obtain the potential feature representation, and good results are obtained by using only prior information. Although the above research studies show that the clustering performance has been significantly improved, they either do not use the supervised information effectively or ignore the unsupervised information.

Apart from the flaws mentioned above, most supervision data used in present deep semi-supervised clustering algorithms is passively collected or arbitrarily selected, and certain approaches require supervision data to suit specific needs. The supervision data chosen randomly or passively gathered may be redundant, and some of it may even hurt the performance of model. More labour is required to supervise information that meets specific needs. Davidson et al. (2006) demonstrated that, when the specified supervision information is unsuitable, the model may eventually produce poor clustering results in semi-supervised clustering methods. As a result, academics are concerned about how to acquire valid supervision data. Some studies have used active learning to pick important supervised information in semi-supervised clustering algorithms and to overcome these difficulties.

Active learning can modify a data set, lowering labour costs, reducing generalization errors, and

https://doi.org/10.24846/v32i3y202301

speeding up model convergence. Combining active learning and the study of deep learning has gained attention in recent years. Most deep active learning algorithms proposed are for classification problems, with only a few for clustering tasks.

A novel deep constrained clustering approach is proposed and a pairwise constrained query strategy is created for the proposed method. The proposed algorithm exploits unlabeled data by self-learning methods and consistent regularization techniques, using pseudo-Siamese networks to learn the relationships between paired data. To obtain more meaningful constraint information, a query strategy is designed based on the estimated confidence of data and the uncertainty estimates of the model.

The main contributions of this paper are the following. A new deep constrained clustering framework is proposed, which uses consistent regularization technology and self-learning to exploit pairwise constraints and unlabeled data. Then, a query strategy for querying paired data in the pseudo-Siamese network is proposed, which combines the traditional query strategy based on confidence query and the model uncertainty strategy used in deep active learning. Finally, extensive tests are conducted on multiple image data sets, and convincing results that verify the reliability of the proposed method are achieved.

The rest of this paper is organized as follows. Section 2 presents the related works. Section 3 provides a detailed description of the proposed method. Section 4 reports the experimental results of the proposed method. The conclusion can be found in Section 5.

2. Related Works

2.1 Deep Constrained Clustering

Due to the rapid advancement of deep learning (Salih et al., 2022; Nath et al., 2022), some scholars are now looking into semi-supervised clustering algorithms combined with deep learning. They use supervised information to guide representation learning and thus obtain more meaningful representations for clustering.

For example, Ren et al. (2019) added a penalty term for pairwise constraints to the deep embedded clustering algorithm (SDEC), which makes the feature space meet the constraints conditions. Arshad et al. (2019) suggested a deep semi-supervised fuzzy c-means clustering algorithm, which is used in multi-category unbalanced classification problems. Ohi et al. (2020) proposed a method for mapping high-dimensional data to embedded systems suitable for clustering (AutoEmbedder), which achieved good clustering performance when using only labeled data. Śmieja et al. (2020) suggested a constrained clustering algorithm using classification ideas (ss-S^3C^2). The procedure is divided into two phases: in the first phase, the Siamese network is trained with pairwise constraints and then the unlabeled paired data is labeled using the trained Siamese network. In the second phase, a new Siamese network is trained with known pairwise constraints and pseudo paired data which was labeled in the first stage.

2.2 Active Semi-Supervised Clustering

Active learning (Ren et al., 2021) can select critical query instances annotated by experts through different query strategies. This process produces a refined data set that reduces labour costs and generalization errors and accelerates model convergence. In semi-supervised clustering methods, active learning has been employed to obtain more semantically supervised informative. Active constrained clustering techniques can be divided into sample-based and sample-pair-based categories, according to Xiong et al. (2016).

The sample-based methods choose the meaningful samples first and then queries the paired data generated according to the selected samples. For example, Basu et al. (2004) first studied the pairwise constraint algorithm based on active learning and proposed the active K-means clustering algorithm FFQS (Farthest-First Query Strategy). They first explored the problem space, query, initialize and augment the sample with known cluster assignments, and after that extracted a substantial number of constraints from the known samples. Xiong et al. (2016) proposed a sample-based active spectral clustering framework that iteratively selects sample-based constraints in an online manner. Each iteration of the algorithm finds the prediction data that minimizes the clustering uncertainty and then uses that data to form pairs for the query.

The sample-pair-based methods select pairs of data to query. For instance, to overcome the

instability of the clustering results caused by the random query, Wang et al. (2020) used an active selection function to query constraint information during clustering in a spectral clustering algorithm. Lutz et al. (2021) proposed an active clustering of training data in which the data is classified by the computer and the pairwise data are then sent to human experts for query. Hazratgholizadeh et al. (2022) proposed an active constrained deep embedding clustering method that utilizes two parallel layers to select information and diversity constraints. Li et al. (2021) proposed an adaptive criteria weights batch selection method that identifies the most informative pairs for semi-supervised clustering through iterative means.

2.3 Siamese Network

A Siamese network is a neural network architecture for learning the similarity between two input data (Anđelić et al., 2021). Bromley et al. (1993) first proposed this approach for signature verification. The main idea is to learn a mapping function from the data making the similarity between two samples of the same class more significant and the similarity between representatives of different categories smaller. The Siamese network consists of two neural networks with the same structure and weight, each capable of learning the hidden representation of the input vector. A Siamese network can be classified as a pseudo-Siamese network if the weights of the two sub-networks are not shared, or if the sub-network architectures differ. Siamese networks are frequently employed to learn the inherent similarity or difference between two objects and have been extensively utilized in various domains such as, natural language processing (Mueller & Thyagarajan, 2016), computer vision (Nandy et al., 2020), and speech processing (Chen & Salman, 2011).

3. Method

3.1 Formulation

For a data set $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathrm{R}^{d \times n}$ consisting of n d-dimensional data, the must-link data set was set as $\mathrm{M} = \{(\mathbf{x}_i, \mathbf{x}_j, \mathrm{y}_{ij} = 1)\}$; and the cannot-link data set was set as $\mathrm{C} = \{(\mathbf{x}_i, \mathbf{x}_j, \mathbf{y}_{ij} = \alpha)\}$, where y_{ij} is a label for pairwise data.

3.2 Overall Framework

To make better use of pairwise constraints and unlabeled data, a novel deep constraint clustering method is introduced. The model uses a pseudo-Siamese network to learn the association between paired data and consistent regularization approaches and self-learning to exploit unlabeled data. Second, a novel query method that uses the confidence of data and the uncertainty of the model-fit and semantically rich supervised information is presented. The framework is divided into two sections: model training and data labeling. Figure 1 illustrates the model structure of the proposed method.

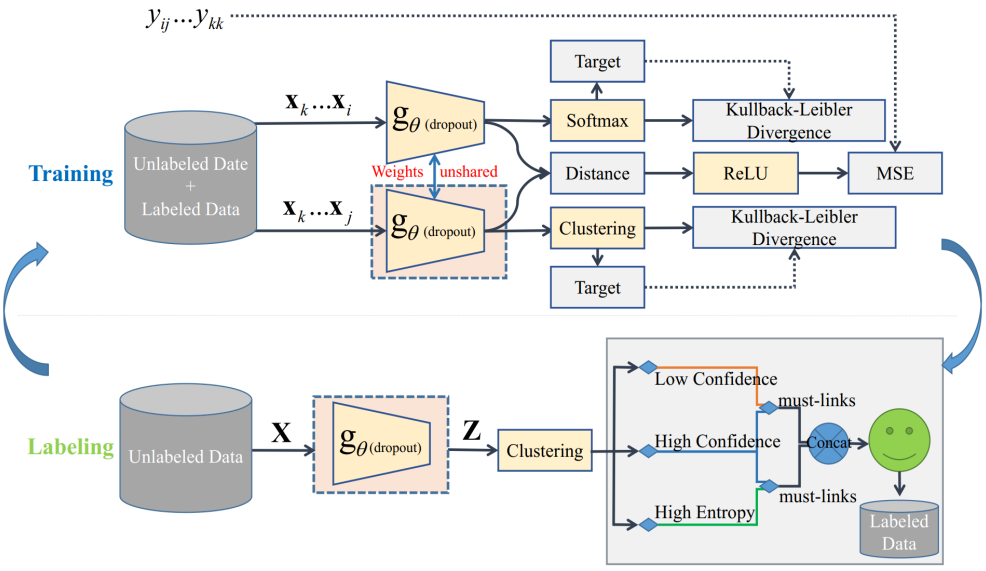

Figure 1. The overview framework of the proposed method

3.3 Model Training Part

In the model training part, a pseudo-Siamese network is composed of a deep neural network with dropout regularization; then a clustering layer and a Dense layer with a Softmax activation function are connected to the two sub-networks of the pseudo-Siamese network. The clustering layer computes the probability assignment of embedded points to cluster centers by Student's *t* distribution. Later, the Dense layer with Softmax function predicts the probability of data belonging to a class. The training of this network involves two phases.

In the training phase, the pseudo-Siamese network is trained using paired data (the pairwise constraints and the must-link pairs composed of unlabeled data). First, the paired data is fed in and a pseudo-Siamese network made up of deep neural networks that have been pre-trained by ImageNet is used to obtain their embedding representations. Second, the Euclidean distance between low-dimensional embedding representations of the paired data is computed. The estimated distance value is then regulated by passing it via a *ReLU* layer. Finally, the network parameters are modified inversely by minimizing the mean squared error loss function between the predicted and actual values.

In the fine-tuning phase, the pseudo-Siamese network is fine-tuned by finding the best fit between the predicted and target distributions. The network parameters are adjusted by minimizing the mean squared error function computed on paired data and the KL divergence loss function calculated on unlabeled data. The prediction distributions in two KL divergence loss functions are the distribution probability of low-dimensional embedded data to the cluster center derived using the Student's *t* distribution and the prediction probability generated using the Softmax function. In contrast, the target distribution is the reinforcement distribution calculated on the prediction distribution.

3.4 Objective Function

One of the loss terms in the proposed technique is a Mean Squared Loss term computed on paired data, while the other two are *KL* divergence loss terms calculated on unlabeled data, namely:

$$L = MSE\,(M,C) + \lambda KL_1\,(\mathbf{X},\mathbf{X}) + \lambda KL_2\,(\mathbf{X},\mathbf{X}) \quad (1)$$

The mean squared error loss is calculated using two types of data: pairwise constraint information and generated must-link pairs of unlabeled data.

For pairwise constraints $(\mathbf{x}_i, \mathbf{x}_j)$, if $y_{ij} = 0$, then $\mathbf{x}_i$ and $\mathbf{x}_j$ are from the same cluster, and if $y_{ij} = \alpha$, then $\mathbf{x}_i$ and $\mathbf{x}_j$ are from different clusters, i.e.,

$$y_{ij} = \begin{cases} 0, \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ must link} \\ \alpha, \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ cannot link} \end{cases} \quad (2)$$

It is worth noting that the outputs of the same sample from the two sub-nets in a pseudo-Siamese network may different due to the usage of dropout regularization and the uncoupling of the weights of the sub-nets. Obviously, $\mathbf{x}_i, \mathbf{x}_i$ belong to the same cluster, and their true label is $y_{ii} = 0$. The network will acquire a low-dimensional embedding $(\mathbf{z}_i, \mathbf{z}'_i)$ for the generated paired data $(\mathbf{x}_i, \mathbf{x}_i)$.

So *MSE*(*M*,*C*) is written as:

$$MSE(M,C) = \frac{1}{|M|+|C|}\sum_{(\mathbf{x}_i,\mathbf{x}_j)\in(M\cup C)} | y_{ij} - f(\mathbf{x}_i,\mathbf{x}_j)|_2^2 \quad (3)$$

where $|.|_2^2$ is the Euclidean norm. $f(\mathbf{x}_i, \mathbf{x}_j)$ is the prediction of paired data, which can be obtained by the distance function *d*() and the hyper-parameter α, i.e.,

$$f(\mathbf{x}_i,\mathbf{x}_j) = ReLU\left(d\left(g_\theta(\mathbf{x}_i), g_{\theta'}(\mathbf{x}_j)\right)\right) = \begin{cases} d(g_\theta(\mathbf{x}_i), g_{\theta'}(\mathbf{x}_j)), \text{if } 0<\mathrm{d}(\mathrm{g}_\theta(\mathbf{x}_i),\mathrm{g}_{\theta'}(\mathbf{x}_j))<\alpha \\ \alpha \qquad\qquad\qquad\quad , \text{if } \mathrm{d}(\mathrm{g}_\theta(\mathbf{x}_i),\mathrm{g}_{\theta'}(\mathbf{x}_j)) \le \alpha \end{cases} \quad (4)$$

$$d\left(g_\theta(\mathbf{x}_i), g_{\theta'}(\mathbf{x}_j)\right) = \left|\mathbf{z}_i - \mathbf{z}_j\right|_2^2 \quad (5)$$

$$\mathbf{z}_i = g_\theta(\mathbf{x}_i) \quad (6)$$

$$\mathbf{z}_j = g_{\theta'}(\mathbf{x}_j) \quad (7)$$

where α is the smallest distance between any two clusters. When the distance between two samples is bigger than α, they are regarded as belonging to different clusters. The nonlinear transformation function *g* transforms high-dimensional data $\mathbf{X}$ into a low-dimensional data $\mathbf{Z}$. θ and θ' represent the model parameter vectors.

KL divergence loss is used to calculate the difference between the predicted distribution Q and the target distribution P of unlabeled data.

$$KL(\mathbf{X},\mathbf{X}) = KL(P\,|\,Q) = \sum_{\mathbf{x}} P(\mathbf{x}) log\frac{P(\mathbf{x})}{Q(\mathbf{x})} \quad (8)$$

The prediction distribution in KL_1 loss is the cluster assignment probability which is calculated by the Student-*t* distribution, i.e.,

$$q_{ij} = \frac{(1+\left|\mathbf{z}_i - \boldsymbol{\mu}_j\right|^2)^{-1}}{\sum_{j'}(1+\left|\mathbf{z}_i - \boldsymbol{\mu}_{j'}\right|^2)^{-1}} \tag{9}$$

where $\mathbf{z}_i$ is the embedding point and $\boldsymbol{\mu}_j$ is the j^{th} cluster center. Then, the final clustering result is obtained by taking the maximum value of cluster allocation q_i, i.e.,

$$c_i = \max_j q_{ij} \tag{10}$$

The prediction distribution in KL_2 loss is the posterior probability which is generated by the sub-net equipped with the Softmax output layer, i.e.,

$$q_{ij} = \frac{e^{\mathbf{z}_j}}{\sum_{j'} e^{\mathbf{z}_{j'}}} \tag{11}$$

On the basis of the predicted distribution, the reinforcement distribution is calculated as the target distribution, i.e.,

$$p_{ij} = \frac{q_{ij}^{\ 2} / f_j}{\sum_{j'} q_{ij}^{\ 2} / f_j} \tag{12}$$

where $f_j = \sum_i q_{ij}$.

3.5 Parameter Optimization

To optimize equation (1), the Adaptive Moment Estimation (ADMA) and back propagation are applied. The formula contains two types of parameters: cluster centers and network parameters. In the fine-tuning phase, the network parameters and cluster centers are collaboratively optimized. And, considering the global structure of data, K-means are utilized for the total data to derive the cluster centers, which are then used to update the cluster center parameters in the clustering layer, after a set number of iterations.

3.6 Parameter Initialization

In the first training phase, the parameters obtained after pre-training the network by ImageNet are used as the initialization parameters of the network. In the second fine-tuning phase, the initial centers μ_j are obtained by applying K-means on the embedding space learned by the sub-net.

3.7 Data Labeling Part

The data labeling part chooses query data from unlabeled data, creates paired data, and sends them to an expert for annotation to update the labeled data set. The paired data whose model predictions are must-link are chosen for the query, because must-link constraints provide more information than cannot-link constraints. There are two types of query paired data: low confidence and high uncertainty. The first object in the low confidence paired data is the low confidence data, while the second object is the high confidence data. The first object in the high uncertainty paired data is high uncertainty data, while the second object is high confidence data. According to the Monte Carlo (MC)-Dropout uncertainty estimation method (Gal & Ghahramani, 2016), the uncertainty measure of the model with respect to the data is obtained by calculating the standard deviation of *T* times forward propagation. The maximum uncertainty estimation probability for each data is taken into account as a measure of uncertainty for the respective data.

3.7.1 Low Confidence Pairs

1. Calculate the confidence level of all data by using equation (10);

2. Select the data with confidence level lower than κ_0 and the data with confidence level higher than κ_1, i.e., $\max_j q_{ij} < \kappa_0$ and $\max_j q_{ij} > \kappa_1$, where κ_0 is the low confidence threshold and κ_1 is the high confidence threshold;

3. Create data pairs using the above-selected data, with the low confidence data as the first object and the high confidence data as the second;

4. Select the data pairs that are predicted by the model as must-link and then give them to the experts for labeling.

3.7.2 High Uncertain Pairs

1. Calculate the confidence level of all data by using equation (10), and calculate the uncertainty of all data as follows:

$$\mathbb{E}(\mathbf{x}_i) \approx \frac{1}{T}\sum_{T=1}^{T} softmax(\mathbf{z}_i) \tag{13}$$

$$Var(\mathbf{x}_i) \approx [softmax(\mathbf{z}_i) - \mathbb{E}(\mathbf{x}_i)]^2 \tag{14}$$

$$u_i = \max_j Var(\mathbf{x}_i) \tag{15}$$

2. Select the data with uncertainty level higher than κ_0 and the data with confidence level higher than κ_1, i.e., $u_i > \kappa_2$ and $\max_j q_{ij} > \kappa$;

3. Create data pairs using the above-selected data, with the high uncertainty data as the first object and the high confidence data as the second;

4. Select the data pairs that are predicted by the model as must-link and then give them to the experts for labeling.

4. Experiments

In order to evaluate the superiority of the proposed method, comparison experiments against some existing unsupervised and semi- supervised methods are conducted in the present paper. Meanwhile, to evaluate the efficacy of the suggested query approach, the clustering performance of the model is examined under different amounts of prior knowledge. Finally, the sensitivity analysis of parameters is also discussed.

4.1 Datasets and Evaluation Metric

The experiments are performed on four image datasets, namely MNIST (Lecun et al., 1998), Fashion (Xiao et al., 2017), Cifar10 (Hull, 1994), and USPS (Krizhevsky & Hinton, 2009).

Table 1 displays the summary information of the datasets.

Table 1. Details of the datasets

Dataset	Classes	Training-size	Testing-size
MNIST	10	60,000	10,000
Fashion	10	60,000	10,000
Cifar10	10	50,000	10,000
USPS	10	9298	9298

Each cluster solution is assessed by using three standard cluster evaluation measures: accuracy (ACC), adjusted rand index (ARI), and normalized mutual information (NMI). The value of ACC and NMI is within the range [0,1], while the value of ARI is within the range [-1,1].

The higher the value of these metrics, the better the clustering results.

The specific definition is as follows.

Accuracy：

$$ACC = \max_m \frac{\sum_{i=1}^{n} \mathbf{1}\{l_i = m(c_i)\}}{n} \tag{16}$$

where l_i is the real label, c_i is assigned to the cluster, and *m* covers the one-to-one correspondence between all the clusters and the label.

Adjusted rand index:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}} \tag{17}$$

where n_{ij}, a_i and b_j are derived from contingency tables.

Normalized mutual information:

$$NMI(c;c') = \frac{I(c;c')}{\max\left(H(c), H(c')\right)} \tag{18}$$

where $I(c;c')$ refers to the mutual information between the true value c and the predicted value c' , and *H*() is the entropy.

4.2 Comparison Methods

Several baseline clustering methods are compared to the approach proposed in this paper. For comparison, unsupervised algorithms such as K-means (MacQueen, 1967) and DEC (Xie et al., 2016) are employed, and semi-supervised clustering approaches such as SDEC (Ren et al. 2019), ss-S^3C^2 (Śmieja et al., 2020), and AutoEmbedder (Ohi et al., 2020) are used.

4.3 Implementation Details

All experimental datasets were divided into training and testing datasets, with the exception of the USPS data set (which was not split). All features are normalized so that they fall between [0,1]. The Pseudo Siamese network's sub-network is composed of MobileNet, Flatten layer, Dropout layer and Dense layer. The parameters of MobileNet are set as: *weights = `ImageNet', include_top = false*, and the rest as defaults. The Dense layer has 64 neurons, and the dropout rate in the Dense layer is set to 0.1. The clustering layer, which computes cluster assignments using the Student-*t* distribution, and the Dense layer,

which uses Softmax as the activation function, are added after the two sub-networks. The Euclidean distance is used to learn the similarity between paired data representations, and the value is limited to [0, α] by the *ReLU* activation function with an upper limit α that can be regarded as the minimum interval between two clusters, which is set to 100.

Since MobileNet requires input data to have three channels and a pixel size of at least 32*32 pixels, MNIST, Fashion, and USPS are scaled to 32*32 pixels and extended to 3-channel data. The optimizer ADMA is used for optimization, and the learning rate is set to 0.001. Each batch of input data contains 128 pairs of paired data, 128 pairs of symmetric forms of paired data, and 128 pairs of paired data consisting of unlabeled data. Except for active learning-based method, all semi-supervised methods had 5,000 pairwise constraints. In the active query strategy, except Cifar10 which is more complex, the initial number of pairwise constraints is set to 500. For more complex dataset as Cifar10, the initial number of pairwise constraints is set to 2,000. All the initial data is selected randomly. Since the maximum number of queries is set to 10, the amount of supervision information constraints of the Active Learning (AL) algorithm on the MNIST and USPS datasets is lower than 2500, while the number of supervision constraints on Fashion data set and Cifar10 is 5000. The low confidence threshold is set as $\kappa_0 = 0.5$ and the high confidence threshold is set as $\kappa_1 = 0.9$. The uncertain threshold is set as $\kappa_2 = 0.8$. MC-Dropout is used to obtain an uncertainty measure, by computing the standard deviation of 10 stochastic forward passes.

5 experiments were conducted on each set of data, and the prior information selected for each experiment was different. Finally, the average of the 5 results was compared with the results from other algorithms.

4.4 Results and Discussions

Table 2 displays the outcomes of the technique suggested in this paper and comparison between the performance evaluations of the present algorithm on the MNIST, Fashion, and USPS data sets. The best values are written in bold. In Table 2, Random denotes that pairwise constraints are chosen randomly, with the exception of the USPS data set, the must-link-to-cannot-link ratio is 1:9, while AutoEmbedder (Balanced) denotes that the ratio should be 1:1. The proposed algorithm is DCC, in which Random has the same meaning as in the brackets above. AL denotes the pairwise constraint information received by the proposed query strategy, where the fraction of must-link and cannot-link is higher than 1:9 and lower than 1:1, respectively.

In Table 2, the deep clustering method DEC outperforms the traditional clustering algorithm K-means in all compared datasets, which proves that increasing deep learning has a positive impact on clustering methods. The clustering performance of SDEC with penalty term added to DEC is improved on all data sets, which means that supervision information does play a guiding role. And DCC is an improved algorithm on AutoEmbedder, which increases the use of unlabeled data. The results indicate that the present method can effectively utilize unlabeled data. But, when the constraint information is randomly selected, except for

Table 2. The clustering ACC, NMI, ARI on MNIST, Fashion and USPS

Method	MNIST			Fashion			USPS		
	ACC	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**	**ACC**	**NMI**	**ARI**
K-means	0.551	0.517	0.385	0.490	0.515	0.350	0.669	0.627	0.546
DEC	0.849	0.816	0.773	0.518	0.546	-	0.758	0.769	0.688
SDEC(Random)	0.861	0.830	0.792	0.440	0.390	0.410	0.764	0.777	0.700
ss-S^3C^2(Random)	0.976	0.928	0.939	0.743	0.373	0.616	0.902	0.899	0.911
AutoEmbedder (Balanced)	0.985	0.958	0.967	0.833	0.774	0.720	0.990	0.973	0.981
DCC-Random	0.991	0.973	0.979	0.846	0.778	0.729	**0.989**	**0.968**	**0.978**
DCC-AL	**0.992**	**0.974**	**0.981**	**0.893**	**0.819**	**0.795**	0.987	0.963	0.974

Cifar10 data set which is more complex, the clustering performance of DCC-Random and AutoEmbedder is comparable in other datasets. This result shows that the value of the amount of information extracted by the proposed algorithm on unlabeled data is less than the value information that must-link brings to the model.

For Cifar10 data set, the performance evaluation results can be seen in Table 3. The clustering performance evaluation results on the prior information obtained by the query strategy are much better than those obtained on the random prior information. However, compared to AutoEmbedder trained with the same number of must-links and cannot-links, the performance of the model proposed in this paper is improved. The main reason is that the information included in must-links is greater than information included in cannot-links. DCC-AL has less expert-annotated must-links than AutoEmbdder.

Figure 2 depicts the change trend of NMI and ARI of the proposed algorithm on MNIST, Fashion and Cifar10 data sets, when query times grow. As it can be seen from Figure 2, the initial accuracy of the model is great for simple datasets MNIST, and the model performance reaches its best sooner. For Fashion data set, the initial model has a fairly high accuracy. The clustering performance of model improves fast as query times increase at the beginning, then the clustering performance of model gradually improves. For the complicated data set Cifar10, the initial accuracy of model is low and the amount of query data has minimal impact on the model in the first few queries. Only when a certain amount of labeled data has been collected does the model accuracy begin to improve.

Table 3. The clustering ACC, NMI, ARI on Cifar10

Method	Cifar10		
	ACC	**NMI**	**ARI**
K-means	0.205	0.085	0.043
AutoEmbedder	0.505	0.404	0.321
DCC-Random	0.444	0.375	0.283
DCC-AL	**0.532**	**0.414**	**0.326**

In general, the proposed query strategy depends on the initial accuracy of the model, for some datasets. Only a small number of queries are needed for the model to perform well, but for complex data more queries are needed.

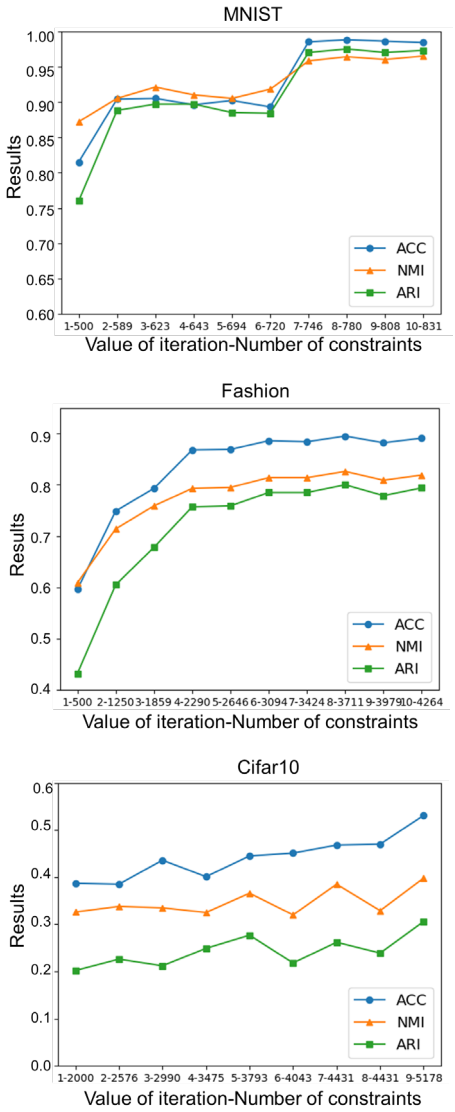

Figure 2. Clustering evaluation results with different pairwise constraints

4.5 Ablation Study

The previous performance comparison of DCC-Random and DCC-AL can prove the effectiveness of the proposed query strategy. This ablation experiment section will look at the effect of consistent training and fine-tuning of the model.

In Table 4, ***Pairwise data*** stands for training the model with pairwise constraints***; Consistency*** stands for training the model using consistent training methods with unlabeled data***; and Consistency + fine-tuning*** stands for using consistent training and fine-tuning techniques.

As it can be seen from Table 4, by comparing the model trained only with pairwise constraints, it becomes clear that the performance of the model with consistency training is greatly improved. By

adding fine-tuning, the performance of the model is also improved, to some extent.

Table 4. The clustering NMI, ARI on Fashion and Cifar10

Method	Fashion		Cifar10	
	NMI	ARI	NMI	ARI
Pairwise data	0.756	0.700	0.380	0.309
Consistency	0.809	0.759	0.494	0.389
Consistency + fine-tuning	**0.821**	**0.780**	**0.505**	**0.391**

4.6 Parameter Analysis

This subsection will discuss three threshold parameters namely κ_0, κ_1, κ_2, in the Data Labeling part of the experiment. Figure 3 shows the clustering results of the model DCC-AL on the Fashion data set with the different values of the low confidence threshold κ_0, the high confidence threshold κ_1, and the high uncertainty threshold κ_2.

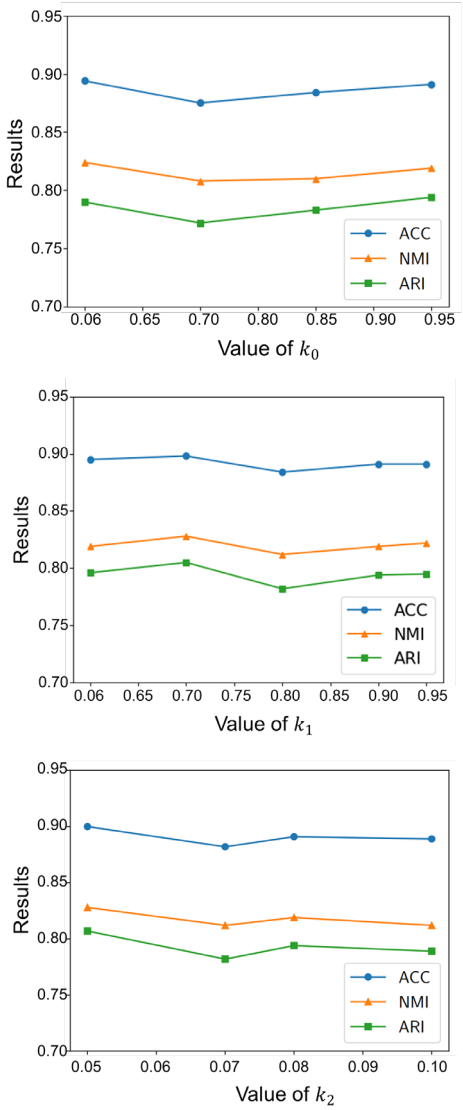

Figure 3. Parameter analysis of Data Labeling part

Only from the results illustrated in Figure 3, the impact that different values of the three parameters have on the performance of the model cannot be clearly seen. However, more must-link data need to be selected in the experiment, so a higher confidence threshold, relatively higher uncertainty thresholds and low confidence thresholds are needed.

5. Conclusion

This paper presents a new method called deep constrained clustering with active learning (DCC-AL), which can solve the issues of limited use of unlabeled and labeled data, as well as low information content and redundancy of a priori knowledge in semi-supervised approaches. The proposed method has been extensively evaluated through numerous experiments, demonstrating its effectiveness.

However, it was found that the initial performance of the model suffers when the data set is too complex, resulting in the fact that the proposed query approach will highlight a lot of data that is not really useful to the model. Furthermore, it was possible to find out in which classes of data the model is more inaccurate by comparing the results of expert labeling with the results predicted by the model. Therefore, in future work, the initial model will be made more accurate for difficult datasets and, furthermore, increasing the weights of class with high error rates will be considered in the query strategy.

REFERENCES

Anđelić, N., Car, Z. & Šercer, M. (2021) Neural network-based model for classification of faults during operation of a robotic manipulator. *Technical Gazette*. 28(4), 1380-1387. doi: 10.17559/TV-20201112163731.

Arshad, A., Riaz, S. & Jiao, L. (2019) Semi-supervised deep fuzzy C-mean clustering for imbalanced multi-class classification. *IEEE Access.* 7, 28100-28112. doi: 10.1109/ACCESS.2019.2901860.

Basu, S., Banerjee, A. & Mooney, R. J. (2004) Active semi-supervision for pairwise constrained clustering. In: *Proceedings of the 4th SIAM International Conference on Data Mining, 22-24 April 2004, Lake Buena Vista, Florida, USA*. Society for Industrial and Applied Mathematics. pp. 333-344.

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E. & Shah, R. (1993) Signature verification using a "Siamese" time delay neural network. In: *Proceedings of the 7th Conference on Neural Information Processing Systems, NeurIPS 1993, 28 November – 1 December 1993, Denver, Colorado, USA.* Burlington, Massachusetts, USA, Morgan Kaufmann. pp. 737-744.

Chen, K. & Salman, A. (2011) Extracting speaker-specific information with a regularized Siamese deep network. In: *Proceedings of the 25th Conference on Neural Information Processing System*s, *NeurIPS 2011, 12-15 December 2011, Granada, Spain*. Neural Information Processing Systems Foundation, Inc. (NeurIPS) - Curran Associates, Inc. pp. 298-306.

Davidson, I., Wagstaff, K. L. & Basu, S. (2006) Measuring constraint-set utility for partitional clustering algorithms. In: *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, PKDD 2006, 18-22 September 2006, Berlin, Germany*. Berlin, Heidelberg, Springer. pp. 115-126.

Gal, Y. & Ghahramani, Z. (2016) Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, 19-24 June 2016, New York, USA*. New York, USA, International Conference on Machine Learning (ICML) - Curran Associates, Inc. pp. 1050-1059.

Hazratgholizadeh, R., Balafar, M. A. & Derakhshi, M. R. F. (2022) Active constrained deep embedded clustering with dual source. *Applied Intelligence*. 53(5), 5337-5367. doi: 10.1007/s10489-022-03752-5.

Hinton, G. E. & Salakhutdinov, R. R. (2006) Reducing the dimensionality of data with neural networks. *Science*. 313(5786), 504-507. doi: 10.1126/science.11276.

Hull, J. J. (1994) A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 16(5), 550-554. doi: 10.1109/34.291440.

Hussein, A., Ahmad, F. K. & Kamaruddin, S. S. (2021) Cluster analysis on Covid-19 outbreak sentiments from twitter data using K-means algorithm. *Journal of System and Management Sciences*. 11(4), 167-189. doi: 10.33168/JSMS.2021.0409.

Jia, Y., Liu, H., Hou, J. & Kwong, S. (2020) Pairwise constraint propagation with dual adversarial manifold regularization. *IEEE Transactions on Neural Networks and Learning Systems.* 31(12), 5575-5587. doi: 10.1109/TNNLS.2020.2970195.

Kim, J. B. (2019) Implementation of artificial intelligence system and traditional system: a comparative study. *Journal of System and Management Sciences*. 9(3), 135-146. doi: 10.33168/JSMS.2019.0309.

Krizhevsky, A. & Hinton, G. (2009) *Learning multiple layers of features from tiny images*. University of Toronto, Toronto, Ontario. Technical report.

Kwon, G. J. & Lee, W. I. (2023) Evolution of innovation clusters from park-type to network-type: focusing on innovation cluster analysis and strategic direction setting. *Journal of Logistics, Informatics and Service Science*. 10(1), 221-236. doi: 10.33168/JLISS.2023.0112.

LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998) Gradient-based learning applied to document recognition. *Proceedings of the IEEE*. 86(11), 2278-2324. doi: 10.1109/5.726791.

Li, H., Wang, Y., Li, Y., Xiao, G., Hu, P., Zhao, R. & Li, B. (2021) Learning adaptive criteria weights for active semi-supervised learning. *Information Sciences*. 561, 286-303. doi: 10.1016/j.ins.2021.01.045.

Lutz, Q., De Panafieu, E., Stein, M. & Scott, A. (2021) Active clustering for labeling training data. In: *Proceedings of the 35th Conference on Neural Information Processing Systems, NeurIPS 2021, 6-14 December 2021, Virtual*. Neural Information Processing Systems Foundation, Inc. (NeurIPS) - Curran Associates, Inc. pp. 8469-8480.

MacQueen, J. (1967). Classification and analysis of multivariate observations. In: *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 21 June – 18 July 1965 and 27 December 1965 – 7 January 1966, Los Angeles, USA*. Berkeley, California, USA, University of California Press. pp. 281-297.

Mueller, J. & Thyagarajan, A. (2016) Siamese recurrent architectures for learning sentence similarity. In: *Proceedings of the 30th AAAI Conference on Artificial Intelligence, 12-17 February 2016, Phoenix, Arizona, USA*. Washington D.C, USA, AAAI Press. pp. 2786-2792.

Nandy, A., Haldar, S., Banerjee, S. & Mitra, S. (2020) A survey on applications of Siamese neural networks in computer vision. In: *Proceedings of 2020 International Conference for Emerging Technology, INCET 2020, 5-7 June 2020, Belgaum, India.* Manhattan, New York City, USA, Institute of Electrical and Electronics Engineers (IEEE) - Curran Associates, Inc. pp. 123-127.

Nath, B., Kumbhar, C. & Khoa, B. T. (2022). A study on approaches to neural machine translation. *Journal of Logistics, Informatics and Service Science.* 9(3), 271-283. doi: 10.33168/LISS.2022.0319.

Nie, F., Zhang, H., Wang, R. & Li, X. (2020) Semi-supervised clustering via pairwise constrained optimal graph. In: *Proceedings of the 29th International Joint Conference on Artificial Intelligence, IJCAI 2020, 7-15 January 2021, Yokohama, Japan*. California, USA, International Joint Conference on Artificial Intelligence (IJCAI) - Curran Associates, Inc. pp. 3160-3166.

Ohi, A. Q., Mridha, M. F., Safir, F. B., Hamid, M. A. & Monowar, M. M. (2020) AutoEmbedder: a semi-supervised DNN embedding system for clustering. *Knowledge-Based Systems*. 204: 106190. doi: 10.1016/j.knosys.2020.106190.

Ren, P., Xiao, Y., Chang, X. Huang, P. Y., Li, Z., Gupta, B. B. & Wang, X. (2021) A survey of deep active learning. *ACM Computing Surveys*. 54(9), 1-40. doi: 10.1145/3472291.

Ren, Y., Hu, K., Dai, X., Pan, L., Hoi, S. C. & Xu, Z. (2019) Semi-supervised deep embedded clustering. *Neurocomputing*. 325, 121-130. doi: 10.1016/j.neucom.2018.10.016.

Safari, S. & Afsari, F. (2020) Ensembling semi-supervised p-spectral clustering for high dimensional data. In: *Proceedings of the 11th Iranian and the First International Conference on Machine Vision and Image Processing, MVIP 2020, 18-20 February 2020, Qom, Iran.* Manhattan, New York City, USA, Institute of Electrical and Electronics Engineers (IEEE) - Curran Associates, Inc. pp. 1-5.

Salih, S., Murat, K., Yanxiao, Z. & Mecit, C. (2022) A comparison of deep learning algorithms on image data for detecting floodwater on roadways. *Computer Science and Information System*. 19(1), 397-414. doi: 10.2298/CSIS210313058S.

Śmieja, M., Struski, Ł. & Figueiredo, M. A. (2020) A classification-based approach to semi-supervised clustering with pairwise constraints. *Neural Networks*. 127, 193-203. doi: 10.1016/j.neunet.2020.04.017.

Wang, X., Ding, S. & Jia, W. (2020) Active constraint spectral clustering based on Hessian matrix. *Soft Computing*. 24, 2381-2390. doi: 10.1007/s00500-019-04069-1.

Xiao, H., Rasul, K. & Vollgraf, R. (2017) Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *ArXiv*. [Preprint] https://arxiv.org/abs/1708.07747 [Accessed 28th July 2023].

Xie, J., Girshick, R. & Farhadi, A. (2016) Unsupervised deep embedding for clustering analysis. In: *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, 19-24 June 2016, New York, USA.* New York, USA, Proceedings of Machine Learning Research (PMLR) (PMLR). pp. 478-487.

Xiong, C., Johnson, D. M. & Corso, J. J. (2016) Active clustering with model-based uncertainty reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 39(1), 5-17. doi: 10.1109/TPAMI.2016.2539965.