A TWO-PHASE EDUCATIONAL DATA CLUSTERING METHOD BASED ON TRANSFER LEARNING AND KERNEL *K*-MEANS

Vo Thi Ngoc Chau, Nguyen Hua Phung

Ho Chi Minh City University of Technology, Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam

Abstract: **In this paper, we propose a two-phase educational data clustering method using transfer learning and kernel *k*-means algorithms for the student data clustering task on a small target data set from a target program while a larger source data set from another source program is available. In the first phase, our method conducts a transfer learning process on both unlabeled target and source data sets to derive several new features and enhance the target space. In the second phase, our method performs kernel *k*-means in the enhanced target feature space to obtain the arbitrarily shaped clusters with more compactness and separation. Compared to the existing works, our work are novel for clustering the similar students into the proper groups based on their study performance at the program level. Besides, the experimental results and statistical tests on real data sets have confirmed the effectiveness of our method with the better clusters.**

Keywords: *Educational data clustering, kernel k-means, transfer learning, unsupervised domain adaptation, kernel-induced Euclidean distance*

I. INTRODUCTION

In the educational data mining area, educational data clustering is among the most popular tasks due to its wide application range. In some existing works [4, 5, 11-13], this clustering task has been investigated and utilized. Bresfelean et al. (2008) [4] used the clusters to generate the student's profiles. Campagni et al. (2014) [5] directed their groups of students based on their grades and delays in examinations to find regularities in course evaluation. Jayabal and Ramanathan (2014) [11] used the resulting clusters of students to analyze the relationships between the study performance and medium of study in main subjects. Jovanovic et al. (2012) [12] aimed to create groups of students based on their cognitive styles and grades in an e-learning system. Kerr and Chung (2012) [13] focused on the key features of student performance based on their actions in the clusters that were discovered. Although the related works have discussed different applications, they all found the clustering task helpful in their educational systems. As for the mining techniques, it is realized that the *k*-means clustering algorithm was popular in most related works [4, 5, 12] while the other clustering algorithms were less popular, e.g. the FANNY algorithm and the AGNES algorithm in [13] and the Partitional Segmentation algorithm in [11]. In addition, each work has prepared and explored their own data sets for the clustering task. There is no benchmark data set for this task nowadays. Above all, none of them has taken into consideration the exploitation of other data sets in supporting their task. It is realized that the data sets in those works are not very large.

Different from the existing works, our work takes into account the educational data clustering task in an academic credit system where our students have a great opportunity of choosing their own learning path. Therefore, it is not easy for us to collect data in this flexible academic credit system. For some programs, we can gather a lot of data while for other programs, we can't. In this paper, a student clustering task is introduced in such a situation. In particular, our work is dedicated to clustering the students enrolled with the target program, called program A. Unfortunately, the data set gathered with the program A is just small. Meanwhile, a larger data set is available with another source program, called program B. Based on this assumption, we define a solution to the clustering task where multiple data sets can be utilized.

As of this moment, a few works such as [14, 20] have used multiple data sources in their mining tasks. However, their mining tasks are student classification [14] and performance prediction [20], not student clustering considered in our work. Besides, [20] was among a very few works proposing transfer learning in the educational data mining area. Voβ et al. (2015) [20] conducted the transfer learning process with Matrix Factorization for data sparseness reduction. It is noted that [20] is different from our work in many aspects: purpose and task. Thus, their approach is unable to be examined in designing a solution of our task.

As a solution to the student clustering task, a two-phase educational data clustering method is proposed in this paper, based on transfer learning and kernel k-means algorithms. In the first phase, our method utilizes both unlabeled target and source data sets in the transfer learning process to derive a number of new features. These new features are from the similarities between the domain-independent features and the domain-specific features in both target and source domains based on spectral clustering at the representation level. They also capture the hidden knowledge transferred from the source data set and thus, help increasing discriminating the instances in the target data set. Therefore, they are the result of the first phase of our method. This result is then used to enhance the target data set where the clustering process is carried out with the kernel k-means algorithm in the second phase of the method. In the second phase, the groups of similar students are formed in the enhanced target feature space so that our resulting groups can be naturally shaped in the enhanced target data space. They are validated with real data sets in comparison with other approaches using both internal and external validation schemes. The experimental results and statistical tests showed that our clusters were significantly better than that from the other approaches. That is we can determine the groups of similar students and also identify the dissimilar students in different groups.

With this proposed solution, we hope that a student clustering task can help educators to group similar students together and further discover the unpleasant cases in our students early. For those in-trouble students, we can provide them with proper consideration and support in time for their final success in study.

The rest of our paper is organized as follows. In section 2, our educational data clustering task is defined. In section 3, we propose a two-phase educational data clustering method as a solution to the clustering task. An empirical study for an evaluation on the proposed method is then given in section 4. In section 5, a review of the related works in comparison with ours is presented. Finally, section 6 concludes this paper and introduces our future works.

II. EDUCATIONAL DATA CLUSTERING TASK DEFINITION

Previously introduced in section 1, an educational data clustering task is investigated in this paper. This task aims at grouping the similar students who are regular undergraduate students enrolled as full-time students of an educational program at a university using an academic credit system. The resulting groups of the similar students are based on their similar study performance so that proper care can go to each student group, especially the group of the in-trouble students who might be facing many difficult problems. Those in-trouble students might also fail to get a degree from the university and thus need to be identified and supported as soon as possible. Otherwise, effort, time, and cost for those students would be wasteful.

Different from the clustering task solved in the existing works, the task in our work is established in the context of an educational program with which a small data set has been gathered. This program is our target program, named program A. On the one hand, such a small data set has a limited number of instances while characterized by a large number of attributes in a very high dimensional space. On the other hand, a data clustering task belongs to the unsupervised learning paradigm where unlike the supervised learning paradigm, only data characteristics are examined during the learning process with no prior information guide. In the meantime, other educational programs, named programs B, have been realized and operated for a while with a lot of available data. These facts lead to a situation where a larger data set from other programs can be taken into consideration for enhancing the task on a smaller data set of the program of interest. Therefore, we formulate our task as a transfer learning-based clustering task that has not yet been addressed in any existing works.

Given the aforesaid purposes and conditions, we formally define the proposed task as a clustering task with the following input and output:

For the input, let D_t denote a data set of the target domain containing n_t instances with $(t+p)$ features in the $(t+p)$-dimensional data vector space. Each instance in D_t represents a student studying the target educational program, i.e. the program **A**. Each feature of an instance corresponds to a subject that each student has to successfully complete to get the degree of the program **A**. Its value is collected from a corresponding grade of the subject. If the grade is not available at the collection time, zero is used instead. With this representation, the study performance of each student is reflected at the program level as we focus on the final study status of each student for graduation. A formal definition is given as follows.

$$D_t = \{X_r, \forall\ r=1..n_t\}$$

where $X_r = (x_{r1}, .., x_{r(t+p)})$ with $x_{rd} \in [0, 10]$, $\forall$ $d=1..(t+p)$

In addition to D_t, let D_s denote a data set of the source domain containing n_s instances with $(s+p)$ features in the $(s+p)$-dimensional data vector space. Each instance in D_s represents a student studying the source educational program, i.e. the program **B**. Each feature of an instance also corresponds to a subject each student has to successfully study for the degree of the program **B**. Its value is also a grade of the subject and zero if not available once collected. D_s is formally defined below.

$$D_s = \{X_r, \forall\ r=1..n_s\}$$

where X_r = (x_{r1}, .., $x_{r(s+p)}$) with $x_{rd} \in$ [0, 10], $\forall$ d=1..(s+p)

In the definitions of D_s and D_t, p is the number of features shared by D_t and D_s. These p features are called pivot features in [3] or domain-independent features in [18]. In our educational domain, they stem from the subjects in common or equivalent subjects of the target and source programs. The remaining numbers of features, t in D_t and s in D_s, are the numbers of the so-called domain-specific features in D_t and D_s, respectively. Moreover, it is worth noting that the size of D_t is much smaller than that of D_s, i.e. $n_t << n_s$.

For the output, the clusters of instances in D_t are returned. Each cluster includes the most similar instances. The instances that belong to different clusters should be dissimilar to each other. Corresponding to each cluster, a group of similar students is derived. These students in the same group share the most similar characteristics in their study performance. In our work, we would like to have the resulting clusters formed in an arbitrary shape in addition to the compactness of each cluster and the separation of the resulting clusters. This implies that the resulting clusters are expected to be the groups of students as natural as possible.

Due to the characteristics of data gathered for the program A, the target program, we would like to enhance the target data set before the processing of the task in the availability of the source data set from program B, the source program. In particular, our work defines a novel two-phase educational data clustering method by utilizing transfer learning in the first phase and performing a clustering algorithm in the second phase. Transfer learning is intended to exploit the existing larger source data set for the more effectiveness of the clustering task on the smaller target data set.

III. THE PROPOSED TWO-PHASE EDUCATIONAL DATA CLUSTERING METHOD

In this section, we propose a two-phase educational data clustering method. This method has two phases. These two phases are sequentially performed. In the first phase, we embed the transfer learning process on both target and source data sets, D_t and D_s, for a feature alignment mapping to derive new features and make a feature enhancement on the target data set D_t. The transfer learning process is defined with normalized spectral clustering at the representation level of both target and source domains. In the second phase, we conduct the clustering process on the enhanced target data set D_t. The clustering process is done with the kernel k-means algorithm. The proposed method results in a transfer learning-based kernel k-means algorithm.

A. Method Definition

The proposed method is defined as follows.

For *the first phase*, transfer learning is conducted on both unlabeled target and source data sets. Based on the ideas and results in [18], transfer learning in our work is developed in a feature-based approach for unsupervised learning in the educational data mining area instead of supervised learning in the text mining area. Indeed, spectral feature alignment in [18] has helped building a new common feature space from both target and source data sets. This common space has been shown for new instances in the target domain to be classified effectively. It implies the significance of the spectral features in well discriminating the instances of the different classes.

Different from [18], we don't align all the features of the target and source domain along with the spectral features in a common space. We also don't build a model on the source data set in the common space and then apply the resulting model on the target data set. For our clustering task, we align only the target features along with the spectral features in the target space so that the target space can be enhanced with new features. Extending a space will help us make the objects apart from each other more. With the new features which are expected to be good for object discrimination, the objects in the enhanced space can be analyzed well for similarity and dissimilarity or for closeness and separation. Therefore, we build a clustering model directly on the target data set in the enhanced space instead of the common space in the second phase.

Because our transfer learning process is carried out on the educational data, the construction of a bipartite graph at the representation level for the texts in [18] can't be considered. Alternatively, we combine the construction steps in [18] and the ones with spectral clustering in [17] for our work. Particularly, our underlying bipartite graph is an undirected weighted graph. In order to build its weight matrix, an association matrix M is first constructed in our work instead of a weight matrix in [18] based on co-occurrence relationships between words. Our association matrix M is based on the association of each domain-specific feature and each domain-independent feature. This association is measured via their similarity with a Gaussian kernel which is somewhat similar to the heat kernel in [2]. The resulting association matrix M is then used to form an affinity matrix A. This affinity matrix A plays a role of an adjacency matrix in spectral graph theory in [7], which is also a weight matrix in [7]. After that, a normalized Laplacian matrix L_N is computed from the affinity matrix A and the degree matrix D for a derivation of the new spectral features.

Based on the largest eigenvalues from eigen decomposition of the normalized Laplacian matrix L_N, a feature alignment mapping is defined with h corresponding eigenvectors. These h eigenvectors form h new spectral features enhancing the target space. In order to transform each instance of the target data set into the enhanced target space, the feature alignment mapping is applied on the target data set.

Regarding parameter settings in the first phase, there are two parameters for consideration: the bandwidth *sigma*$_1$ in the Gaussian kernel and the number *h* of the new spectral features in the enhanced space. After examining the heat kernel in [2], we realized that *sigma*$_1$ is equivalent to *t*, which was stated to have little impact on the resulting eigenvectors. On the other hand, in [17], *sigma*$_1$ was checked in a grid search scheme to have an automatic setting for spectral clustering. In our work, spectral clustering is for finding new features in the common space of the target and source domains and thus, not directly associated with the ultimate clusters. Hence, we decide to automatically derive a value for *sigma*$_1$ from the variances in the target data set. Variances are included because of their averaged standard differences in data. In addition, the target data set is considered instead of both target and source data sets because of feature enhancement on the target space, not on the common space. Different from the first parameter *sigma*$_1$, the second parameter *h* gives us the extent of the hidden knowledge transferred from the source domain. What value is proper for this parameter depends on the source data set that has been used in transfer learning. It also depends on the relatedness of the target domain and source domain via the domain-independent feature set on which the new common space is based. Therefore, in our work, we don't derive any value for the parameter h automatically from the data sets. Instead, its value is investigated with an empirical study in particular domains.

For *the second phase*, kernel *k*-means is performed on the enhanced target data set. Different from the existing kernel *k*-means algorithms as described in [19], kernel *k*-means used in our work is defined with three following points for better effectiveness.

Firstly, we establish the objective function in the feature space based on the enhanced target space instead of the original target space. That is we have counted the new spectral features in the feature space so that the implicit knowledge transferred from the source domain can help the clustering process discriminate the instances. The following is the objective function in our kernel *k*-means clustering process in the feature space with an implicit mapping function Φ. This function value is minimized iteration by iteration till the clusters can be shaped firmly.

$$J^{\Phi}(D_t, C^{\Phi}) = \sum_{r=1..n_t} \sum_{o=1..k} \gamma_{or} \| \Phi(X_r) - C_o \|^2 \tag{1}$$

Where $X_r = (x_{r1}, .., x_{r(t+p)}, \varphi(X_r))$ is an instance in the enhanced target space. γ_{or} is the membership of X_r with respect to the cluster whose center is C_o: 1 if a member and 0 if not. C_o is a cluster center in the feature space with an implicit mapping function Φ, defined as follows.

$$C_o = \frac{\sum_{q=1..n_t} \gamma_{oq} \Phi(X_q)}{\sum_{q=1..n_t} \gamma_{oq}} \tag{2}$$

Using the kernel matrix with the Gaussian kernel function, the corresponding objective function is computationally defined with an implicit mapping function Φ as follows.

$$J^{\Phi}(D_t, C^{\Phi}) = \sum_{r=1..n_t} \sum_{o=1..k} \gamma_{or} \left(K_{rr} - \frac{2 \sum_{q=1..n_t} \gamma_{oq} K_{rq}}{\sum_{q=1..n_t} \gamma_{oq}} + \frac{\sum_{v=1..n_t} \sum_{z=1..n_t} \gamma_{ov} \gamma_{oz} K_{vz}}{\sum_{v=1..n_t} \sum_{z=1..n_t} \gamma_{ov} \gamma_{oz}} \right) \tag{3}$$

Where γ_{or}, γ_{oq}, γ_{ov}, and γ_{oz} are memberships of the instances X_r, X_q, X_v, and X_z with respect to the cluster whose center is C_o. In the kernel matrix, we can have K_{rr}, K_{rq}, and K_{vz} computed below:

$$K_{rr} = e^{-\frac{\| X_r - X_r \|^2}{2 * sigma_2{}^2}} = 1 \tag{4}$$

$$K_{rq} = e^{-\frac{\| X_r - X_q \|^2}{2 * sigma_2{}^2}} \tag{5}$$

$$K_{vz} = e^{-\frac{\| X_v - X_z \|^2}{2 * sigma_2{}^2}} \tag{6}$$

Where each Euclidean distance between the instances is computed in the enhanced target space rather than the original target space.

$$\| X_r - X_q \| = \sqrt{\sum_{d=1..(t+p+h)} (x_{rd} - x_{qd})^2} \tag{7}$$

$$\| X_v - X_z \| = \sqrt{\sum_{d=1..(t+p+h)} (x_{vd} - x_{zd})^2} \tag{8}$$

Secondly, we derive a value for the bandwidth parameter *sigma*$_2$ of the kernel function automatically from the variances in data instead of asking the users for a proper value. The foundation of this derivation is based on the meaning and use context of the kernel function value. In theory, if the kernel function is a covariance function used in Gaussian processes, then the kernel matrix can be a covariance matrix. Besides, in our clustering process, the kernel matrix computed with the Gaussian kernel function is used for computing distances between the instances and the cluster centers in the feature space. Generally speaking, the bandwidth parameter *sigma*$_2$ scales the distances between two objects in the enhanced target space before it is considered in the feature space. If *sigma*$_2$ is so small, the distances between two objects in the feature space will get constant and thus, unable to discriminate between the instances. If *sigma*$_2$ is so large, the distances between two objects in the feature space will get close to that in the data space. Both cases have an impact on the resulting clusters. In our work, *sigma*$_2$ is determined automatically from the variances in the target data set so that the differences between the instances to be clustered can be considered in the mapping of the instances between the data and feature spaces.

Thirdly, we reduce the randomness in initialization of initial clusters in kernel *k*-means by using the clusters resulted in *k*-means in the enhanced target space. The *k*-means clustering process provides us with the draft partition of the enhanced target space. Therefore, initialization with the clusters from *k*-means has little difference from execution to execution as compared to initialization with completely random clusters. Such a choice makes our method more stable while increases the computational cost little because *k*-means is one of the algorithms with the smallest computational cost.

As for the convergence of kernel *k*-means, no change in the clusters formed so far will signal for the stability of the clustering process. We use this status as a termination condition. The resulting clusters in the feature space are in hyper-spherical shapes and thus, in non-hyper-spherical shapes in the data space when we derive the membership of each instance with respect to the resulting clusters in the data space. This fact helps us achieving the clusters of higher quality as compared to that from the original *k*-means algorithm.

Corresponding to the aforementioned method definition, the pseudo code of the resulting transfer learning-based kernel *k*-means algorithm is given in Algorithm I.

Algorithm I: The proposed transfer learning-based kernel k-means algorithm

```
Algorithm: Transfer learning-based kernel k-means
Input:
        $D_t$: a data set of the target domain containing $n_t$ instances
        $D_s$: a data set of the source domain containing $n_s$ instances
        $t$: the number of features of the target domain, called domain-specific features
        $s$: the number of features of the source domain, called domain-specific features
        $p$: the number of features in common of both source and target domains, called domain-independent features
        $h$: the number of enhanced features
        $k$: the number of clusters
Output: $k$ clusters with the cluster centers such that C = {$C_1$, $C_2$, .., $C_k$}
Process:
Phase 1 - Derive h enhanced features
1.1. Construct an association matrix $M$ showing the association of each domain-specific feature and each
     domain-independent feature:
                $M = [m_{ij}]$ for $i=1..p$ and $j=1..(s+t)$                                        (9)
where each $i$-th and $j$-th cell of $M$ is calculated as follows:
                $m_{ij} = e^{\frac{-\|A_i - A_j\|}{2*sigma_1^2}}$                                   (10)
where $\|A_i-A_j\|$ is used for measuring the similarity between a domain-independent feature $A_i$ and a domain-
specific feature $A_j$ via a Euclidean distance in the data space of each domain:
                $\| A_i - A_j \| = \sqrt{\sum_{r=1..n} (x_{ri} - x_{rj})^2}$                        (11)
where $n$ is the number of source/target instances, i.e. $n=n_s$ for domain-specific features in $D_s$ and $n=n_t$ for
domain-specific features in $D_t$.
In our method, the Gaussian function is used with $sigma_1$ automatically derived from the variances in the data
of $D_t$.
                $sigma_1 = 0.3 * \sum_{r=1..n_t} \frac{1}{t+p} \left( \sum_{d=1..(t+p)} x_{rd}^2 - \frac{1}{t+p} \left( \sum_{d=1..(t+p)} x_{rd} \right)^2 \right)$     (12)
```

1.2. Form an affinity matrix *A*:

$$A = \begin{bmatrix} 0 & M \\ M^T & 0 \end{bmatrix} \qquad (13)$$

where M^T is a transpose of the association matrix *M*.

1.3. Compute the normalized Laplacian matrix L_N:

$$L_N = [nl_{ij}] \text{ for } i=1..(s+t+p), j=1..(s+t+p) \qquad (14)$$

$$nl_{ij} = A_{ij} /(\sqrt{D_{ii}} * \sqrt{D_{jj}}) \qquad (15)$$

where A_{ij} is the *i*-th and *j*-th cell in the affinity matrix *A* and the degree matrix *D* which is a diagonal matrix with:

$$D_{ii} = \sum_{j=1..(s+t+p)} A_{ij} \text{ for } i=1..(s+t+p) \qquad (16)$$

1.4. Find *h* eigenvectors of L_N: u_1, u_2, …, u_h that are associated with the *h* largest eigenvalues

1.5. Form the transformation matrix *U*:

$$U = [u_1 u_2 \ldots u_\mathrm{h}] \qquad (17)$$

1.6. Derive *h* enhanced features for each instance $X_r = (x_{r1}, .., x_{r(t+p)})$ in D_t for r=1..n_t by means of a feature alignment mapping $\varphi(X_r)$:

$$\varphi(X_r) = (x_{r1}, .., x_{r(t+p)}, 0, \ldots, 0)*U \qquad (18)$$

where (0, …, 0) is a zero placeholder for *s* source-specific features in the mapping. Each instance X_r is returned as: $(x_{r1}, .., x_{r(t+p)}, \varphi(X_r))$.

Phase 2 - Generate k clusters in the enhanced target feature space where D_t is enhanced

2.1. Compute the kernel matrix *KM* each cell of which is calculated using the Gaussian function:

$$KM(X_r, X_q) = K_{rq} = e^{-\frac{\|X_r - X_q\|^2}{2*sigma_2{}^2}} \qquad (19)$$

where $\|X_r\text{-}X_q\|$ is a Euclidean distance between two instances X_r and X_q in the data space:

$$\| X_r - X_q \| = \sqrt{\sum_{d=1..(t+p+h)} (x_{rd} - x_{qd})^2} \text{ for } r=1..n_t \text{ and } q=1..n_t \qquad (20)$$

and $sigma_2$ is derived automatically from the variances in the data of D_t

$$sigma_2 = 0.3* \sum_{d=1..(t+p+h)} \mathrm{var}_d \qquad (21)$$

$$\mathrm{var}_d = \frac{1}{n_t}\Big(\sum_{r=1..n_t} x_{rd}^2 - \frac{1}{n_t}\big(\sum_{r=1..n_t} x_{rd}\big)^2\Big) \text{ for } d = 1..(t+p+h) \qquad (22)$$

2.2. Initialize the cluster centers from *k* resulting clusters of the standard *k*-means algorithm on the target data set D_t.

2.3. Repeat the following actions 2.4 and 2.5 until the membership of each instance is unchanged in the feature space, i.e. the value of the objective function is unchanged.

2.4. Update the distance between each cluster center C_o and each instance X_r in the feature space for *o*=1..*k* and *r*=1..n_t

$$\| \phi(X_r) - C_o \|^2 = K_{rr} - \frac{2 \sum_{q=1..n_t} \gamma_{oq} K_{rq}}{\sum_{q=1..n_t} \gamma_{oq}} + \frac{\sum_{v=1..n_t} \sum_{z=1..n_t} \gamma_{ov}\gamma_{oz} K_{vz}}{\sum_{v=1..n_t} \sum_{z=1..n_t} \gamma_{ov}\gamma_{oz}} \qquad (23)$$

where γ_{oq}, γ_{ov}, and γ_{oz} are the current memberships of the instances X_q, X_v, and X_z with respect to the cluster center C_o.

$$\gamma_{oq} = \begin{cases} 1, \text{if } X_q \text{ is a member of } C_o \\ 0, \text{otherwise} \end{cases}$$

$$\gamma_{ov} = \begin{cases} 1, \text{if } X_v \text{ is a member of } C_o \\ 0, \text{otherwise} \end{cases}$$

$$\gamma_{oz} = \begin{cases} 1, \text{if } X_z \text{ is a member of } C_o \\ 0, \text{otherwise} \end{cases}$$

2.5. Update the membership γ_{oq} between the instance X_r and the cluster center C_o for r=1..n_t and o=1..k

$$\gamma_{oq} = \begin{cases} 1, \text{if } \|\phi(X_r) - C_o\|^2 = \text{argmin}_{o'=1..k}(\|\phi(X_r) - C_{o'}\|^2) \\ 0, \text{otherwise} \end{cases} \quad (24)$$

2.6. Return k clusters based on the membership of each instance with respect to each cluster.

B. Characteristics of the Proposed Method

As described above, the proposed method is a novel solution for educational data clustering in the context where the target domain has a small data set for the task. The method conducts the transfer learning process on both target and source data sets for the new features that can enhance the target data space for better instance discrimination. The method then performs the clustering process on the target data set in the enhanced target feature space with kernel-induced distances. It is worth noting that our method has no execution of the clustering process on the source data set. Such a design helped us save a lot of computational cost because in the context of our clustering task, the source data set is much larger than the target data set.

Different from the existing transfer learning-based clustering approach, self-taught clustering in [8], our approach exploited the source data set at the representation level while Dai et al. (2008) [8]'s approach at the instance level. In addition, our approach did not perform the clustering process on the source data set while Dai et al. (2008) [8]'s approach required the clustering process on both source and target data sets. As based on the kernel k-means algorithm, our approach aimed at the clusters in the feature space instead of in the data space as considered in [8].

As compared to [16], our transfer learning approach is considered at the representation level while that in [16] at the instance level. Martín-Wanton et al. (2013) [16] defined their unsupervised transfer learning method using Latent Dirichlet Allocation (LDA) for short text clustering. The method was run on both target and source data sets and then derived the clusters of the target data set by removing the source instances in the resulting clusters containing at least one target instance. This method assumed that the source and target domains shared the same space. This assumption is relaxed in our method where there exist domain-specific features.

Different from the existing approaches to educational data clustering in [4, 5, 12], our method was based on the kernel k-means clustering algorithm while [4, 5, 12]'s methods were based on the k-means clustering algorithm. We believe that the student groups created from our method are of higher quality as non-linearly formed in the enhanced target data space. In addition, our method not only used one target data set but also exploited another source data set for better representation.

In short, our work has defined a new transfer learning-based clustering approach in the educational domain. The resulting two-phase clustering method is expected to produce the clusters of higher quality in more natural shapes. This method is also a novel solution for grouping similar students based on their study performance at the program level.

IV. EVALUATION

For an evaluation of the proposed method, we conducted an empirical study with many experiments and numerical analysis in this section.

A. Data and Experiment Settings

In this work, we have implemented the proposed method in Matlab and Java: the first phase with Matlab and the second phase with Java. The resulted data after feature enhancement in the first phase are organized in the .csv files which are then processed by the kernel k-means clustering algorithm in the second phase. With that implementation, our experiments were carried out on a 2.2 GHz Intel Core i7 notebook with 6.00 GB RAM running Windows 7 Ultimate, a 64-bit operating system.

As previously mentioned in the educational data clustering task definition, our target data set is so smaller than other available source data set in the education domain. Indeed, our target data set contains 186 instances stemming from the program in Computer Engineering (CE), i.e. the program A, and our source data set consists of 1317 instances from the program in Computer Science (CS), i.e. the program B. These two data sets are real data sets from grade information of the corresponding undergraduate students enrolled in 2008-2009 for the program A and in 2005-2008 for the program B, both in the academic credit system at Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam, [1].

Table I. Details of data sets

Educational Program	Study Years	Student #	Class #	Feature #	Common Feature #
CE (Target, A)	2, 3, 4	186	3	43	32
CS (Source, B)	2, 3, 4	1,317	3	43	

For being used in comparisons, three data sets were built for each program corresponding to 3 years of study of the aforementioned students from year 2 to year 4: Year 2, Year 3, Year 4. The Year 2 data set is with the 2nd-year students, the Year 3 data set with the 3rd-year students, and the Year 4 data set with the 4th-year students for both programs. Their details are briefly described in Table I.

Each instance in a data set in both programs has 43 attributes originally corresponding to 43 subjects and 1 class attribute whose values are either “graduating”, “studying”, or “study_stop” corresponding to the final study status of a student. We prepared the class attribute for external validation. In the real world, two programs have 32 subjects in common which are mainly basic subjects for general knowledge and English as well as fundamental subjects for core knowledge in the computer field. These 32 subjects form 32 common features between two domains: target and source. They are so-called pivot features in [3] and so-called domain-independent features in [18]. The remainder subjects form domain-specific features of each domain.

As for the processing in the first phase of the proposed method, different feature spaces are considered in this evaluation: original and enhanced. The original space is the one that we have described above. The enhanced space is the one that we have obtained with transfer learning between these two programs using spectral clustering. This enhanced space is generated by adding several enhanced features from transfer learning to the original space. Different numbers of enhanced features are examined starting from the number of classes to the higher numbers: {3, 6, 9, 12, 15} corresponding to {*k*, 2**k*, 3**k*, 4**k*, 5**k*}. The reported results of the algorithms are based on the stability of the changes in validity indices. In addition, the bandwidth $sigma_1$ of the Gaussian kernel in the transfer learning process is automatically determined from the variance of the target data set as proposed. For evaluation, we reported the results with different values for $sigma_1$: 0.03*sum_of_variances_1, 0.3*sum_of_variances_1, 3*sum_of_variances_1, 30*sum_of_variances_1, and 300*sum_of_variances_1 where sum_of_variances_1 (var1 for short) is derived from the total sum of the variance for each instance in the target data.

For the clustering algorithms in the second phase, the original *k*-means and kernel *k*-means algorithms in the original data space were used for comparison. The number of clusters *k* is chosen from the number of classes of the data sets for both *k*-means and kernel *k*-means algorithms. It is set to 3. As for the kernel *k*-means algorithm, the kernel function is the Gaussian kernel function for its capability of non-linear transformation. As earlier proposed, the bandwidth $sigma_2$ of the kernel is automatically determined from the variance of each data set. For evaluation, different values for $sigma_2$ are considered: 0.03*sum_of_variances_2, 0.3*sum_of_variances_2, 3*sum_of_variances_2, 30*sum_of_variances_2, and 300*sum_of_variances_2 where sum_of_variances_2 (var2 for short) is derived from the total sum of the variance for each attribute in the target data.

For randomness avoidance in initialization, we used the same initial values for the clustering algorithms. In addition, 100 runs were carried out for each experiment. Averaged results are then recorded. Their standard deviations are also derived and displayed.

For validation of the resulting clusters in each experiment, two validation schemes were examined: internal and external. For internal validation, three well-known measures used are: Objective Function, S_Dbw, and Dunn. Objective Function is used for checking the optimization of the partitioning approaches. Both S_Dbw and Dunn are used for examining the separation and compactness of the resulting clusters; but S_Dbw is more preferred with respect to monotonicity, noise, density, subclusters, and skewed distributions in data as discussed in [15]. For external validation, Entropy is used for its simplicity and popularity toward supervised learning. For better resulting clusters, we expect smaller values of Objective Function, S_Dbw, and Entropy and larger values of Dunn. More computing details of these measures can be found in [15, 21].

For checking significant differences in comparison, One-Way ANOVA was conducted with equal variances assumed for post hoc multiple comparisons with Bonferroni, LSD, and Tukey HSD at the 0.05 level of significance. Levene Statistic is also included for a test of homogeneity of variances. The case of 15 enhanced features for all the data sets is used in statistical tests. All the statistical tests show the differences between the results from the proposed method and that from the others are significant.

B. Experimental Results and Discussions

In this subsection, we present the experimental results in two main groups: the first group for a study of the effectiveness of our method and the second for a study of the affect of the parameters in our method.

In the first group, Table II and Table III give us the averaged results and their standard deviations, respectively, for two clustering algorithms *k*-means and kernel *k*-means on original data sets and enhanced data sets. In this group, we used 15 enhanced features, $sigma_1$ = 0.3*var1, and $sigma_2$ = 0.3*var2. The best averaged results are displayed in bold. It is realized that the proposed method with the kernel *k*-means clustering algorithm on the enhanced data sets outperforms the other methods, such as the methods with the *k*-means clustering algorithm on either original or enhanced data sets and with the kernel *k*-means clustering algorithms on the original data sets. The effectiveness of the proposed method is reached on a consistent basis via all the measures: Objective Function, S_Dbw, Dunn, and Entropy. In addition, standard deviations in Table III are small values for the measures S_Dbw, Dunn, and Entropy and quite large values for the measure Objective Function. The Objective Function values of the proposed method are among the smallest one for standard deviations, showing the stability of the proposed method in its convergence as compared to those of the others.

*Table II. Average results of 100 runs with original data sets and enhanced data sets with the number of enhanced features = 15, $sigma_1$ = 0.3*var1, and $sigma_2$ = 0.3*var2*

Data set	Feature space	Method	Objective Function	S_Dbw	Dunn	Entropy
Year 2	Original	*k*-means	530.04	0.81	0.16	1.13
	Enhanced	*k*-means	389.20	0.80	0.15	1.13
	Original	Kernel *k*-means	483.97	0.78	**0.18**	1.01
	Enhanced	Kernel *k*-means	**347.57**	**0.75**	**0.18**	**0.98**
Year 3	Original	*k*-means	601.25	0.73	0.16	1.01
	Enhanced	*k*-means	447.09	0.70	0.16	1.00
	Original	Kernel *k*-means	538.88	0.71	0.17	0.86
	Enhanced	Kernel *k*-means	**398.03**	**0.67**	**0.19**	**0.84**
Year 4	Original	*k*-means	749.76	0.62	0.16	0.98
	Enhanced	*k*-means	604.80	0.52	0.15	0.93
	Original	Kernel *k*-means	641.45	0.58	**0.19**	0.85
	Enhanced	Kernel *k*-means	**505.58**	**0.46**	**0.19**	**0.81**

*Table III. Standard deviations of 100 runs with original data sets and enhanced data sets with the number of enhanced features = 15, $sigma_1$ = 0.3*var1, and $sigma_2$ = 0.3*var2*

Data set	Feature space	Method	Objective Function	S_Dbw	Dunn	Entropy
Year 2	Original	*k*-means	53.38	0.08	0.04	0.13
	Enhanced	*k*-means	47.31	0.08	0.04	0.14
	Original	Kernel *k*-means	27.77	0.06	0.04	0.08
	Enhanced	Kernel *k*-means	25.96	0.05	0.04	0.09
Year 3	Original	*k*-means	74.46	0.09	0.06	0.13
	Enhanced	*k*-means	61.31	0.09	0.06	0.14
	Original	Kernel *k*-means	36.63	0.07	0.05	0.09
	Enhanced	Kernel *k*-means	28.26	0.06	0.06	0.08
Year 4	Original	*k*-means	148.13	0.10	0.06	0.16
	Enhanced	*k*-means	145.07	0.10	0.06	0.15
	Original	Kernel *k*-means	64.82	0.06	0.06	0.08

	Enhanced	Kernel *k*-means	43.51	0.04	0.06	0.10

*Table IV. Average results of 100 runs of the kernel k-means method with data sets with different numbers of enhanced features while fixing sigma$_1$ = 0.3*var1 and sigma$_2$ = 0.3*var2*

Data set	Enhanced Feature#	Objective Function	S_Dbw	Dunn	Entropy
Year 2	0	483.97	0.78	0.18	1.01
	3	441.31	0.45	0.16	0.96
	6	417.88	0.54	0.16	0.96
	9	386.82	0.74	0.17	0.98
	12	361.08	0.75	0.18	0.97
	15	347.57	0.75	0.18	0.98
Year 3	0	538.88	0.71	0.17	0.86
	3	553.32	0.31	0.13	0.80
	6	505.82	0.43	0.16	0.81
	9	451.10	0.57	0.17	0.82
	12	416.42	0.64	0.18	0.82
	15	398.03	0.67	0.19	0.84
Year 4	0	641.45	0.58	0.19	0.85
	3	846.32	0.19	0.11	0.76
	6	696.03	0.25	0.15	0.78
	9	621.08	0.31	0.16	0.81
	12	564.71	0.39	0.17	0.79
	15	505.58	0.46	0.19	0.81

In the second group, Tables IV-VI present the average results of 100 runs with the kernel *k*-means algorithm with different settings in the proposed method. Particularly, Table IV is for different numbers of enhanced features and *sigma$_1$* = 0.3*var1 and *sigma$_2$* = 0.3*var2, Table V is for different values of *sigma$_1$* and the number of enhanced features = 15 and *sigma$_2$* = 0.3*var2, and Table VI is for different values of *sigma$_2$* and *sigma$_1$* = 0.3*var1 and the number of enhanced features = 15. Changes in the number of enhanced features and *sigma$_1$* are considered for transfer learning to capture the similarity in the source space and the target space via spectral clustering while changes in the number of *sigma$_2$* is considered for kernel clustering to make non-linear transformation between the data space and the feature space via kernel-induced distances. It is figured out that different numbers of enhanced features are linked to different averaged results significantly in Table IV while different values of *sigma$_1$* and *sigma$_2$* in Tables V and VI have no significant difference in averaged results of the measures: Objective Function, S_Dbw, Dunn, and Entropy. This leads to an appropriateness of the settings in our proposed method. Indeed, deriving *sigma$_1$* and *sigma$_2$* automatically from the variances in the target data set is applicable with little impact on the final results so that the proposed method can be directed to a parameter-free version. This also makes the proposed method more practical from the user's side. As a result, users are only asked for the number of clusters and the number of enhanced features. The first parameter is related to a typical issue with the partitioning approach while the second one to a typical issue with feature space enhancement based on transfer learning. As for the number of enhanced features, shown in Table IV, the best results for the measures S_Dbw, Dunn, and Entropy are associated with 3 enhanced features while the best results for Objective Function with 15 enhanced features. Nevertheless, the stability of the proposed method increases as the number of enhanced features increases in spite of not the best results. As displayed in Table II for comparison with different methods, the proposed method still produces better results even with 15 enhanced features. This fact shows an appropriateness of the proposed method using the kernel *k*-means algorithm in the enhanced feature space.

In short, it is found that our two-phase clustering method is effective with a combination of spectral clustering for transfer learning between two domains and kernel *k*-means for clustering similar transformed instances in the feature space for educational data clustering. As a result, our algorithm can generate several groups of the most similar instances in spherically-shaped clusters in the enhanced feature space which are actually arbitrarily-shaped clusters in the enhanced data space. Those clusters are compact and well-separated as confirmed with the better averaged results of both external and internal validity measure groups: (Entropy) and (Objective Function, S_Dbw, Dunn), respectively. Statistical tests also show that all the better differences between our proposed method and the others are statistically significant at the significance level of 0.05.

*Table V. Average results of 100 runs of the kernel k-means method with data sets with different values of sigma$_1$ while fixing the number of enhanced features = 15 and sigma$_2$ = 0.3*var2*

Data set	***sigma$_1$***	**Objective Function**	**S_Dbw**	**Dunn**	**Entropy**
Year 2	0.03*var1	345.84	0.75	0.18	0.98
	0.3*var1	347.57	0.75	0.18	0.98
	3*var1	345.14	0.75	0.19	0.98
	30*var1	345.25	0.75	0.17	0.98
	300*var1	343.96	0.75	0.18	0.98
Year 3	0.03*var1	397.44	0.67	0.17	0.83
	0.3*var1	398.03	0.67	0.19	0.84
	3*var1	396.88	0.68	0.18	0.83
	30*var1	393.56	0.68	0.18	0.81
	300*var1	404.43	0.67	0.17	0.84
Year 4	0.03*var1	511.07	0.48	0.19	0.81
	0.3*var1	505.58	0.46	0.19	0.81
	3*var1	503.54	0.48	0.19	0.79
	30*var1	512.31	0.47	0.17	0.80
	300*var1	508.74	0.47	0.18	0.82

*Table VI. Average results of 100 runs of the kernel k-means method with data sets with different values of sigma$_2$ while fixing the number of enhanced features = 15 and sigma$_1$ = 0.3*var1*

Data set	***sigma$_2$***	**Objective Function**	**S_Dbw**	**Dunn**	**Entropy**
Year 2	0.03*var2	367.95	0.78	0.17	1.04
	0.3*var2	347.57	0.75	0.18	0.98
	3*var2	344.38	0.75	0.18	0.97
	30*var2	344.57	0.75	0.18	0.97
	300*var2	342.71	0.75	0.19	0.98
Year 3	0.03*var2	434.47	0.67	0.16	0.94
	0.3*var2	398.03	0.67	0.19	0.84
	3*var2	401.13	0.67	0.18	0.84
	30*var2	396.87	0.66	0.18	0.83
	300*var2	402.14	0.68	0.18	0.84
Year 4	0.03*var2	561.07	0.48	0.18	0.84
	0.3*var2	505.58	0.46	0.19	0.81
	3*var2	509.70	0.46	0.18	0.81
	30*var2	514.25	0.46	0.18	0.81

	300*var2	511.29	0.48	0.18	0.82

V. RELATED WORKS

Transfer learning has been received much attention in the multimedia data mining area such as text mining and image mining areas. Among the first works bringing transfer learning to the educational data mining area, our work aims to obtain better mining models from smaller data sets with a great support of transfer learning. In this review on the related works, we figure out the design rationale of our method and compare it with the ones in the related works.

Firstly, several works on domain adaptation such as [3, 6, 9, 10, 18, 22] are discussed. Most of the works on domain adaptation were dedicated to data classification and that implies few of them for data clustering. Therefore, their transfer learning process often exploited the labeled source data sets. They are detailed as follows.

Indeed, Chang et al. (2017) [6] used both labeled data in the source and target domain for the training phase. Different from the existing works, Chang et al. (2017) [6] has relaxed the use of domain-independent features. Instead of them for a shared space between the target and source domains, Chang et al. (2017) [6] requested a parallel data set which plays a role of a bridge to create the relations between the target and source domains. In our educational context, it is not trivial to reach such a parallel data set. Duan et al. (2012) [9] proposed the Heterogeneous Feature Augmentation (HFA) method to derive new augmented feature representations for learning tasks. Their method didn't ask for the optimal dimension of the common subspace. However, the method required learning the projection matrices via learning the transformation metric. In addition, Feuz and Cook (2015) [10] defined the feature space remapping method for two cases: unlabeled and labeled target data sets, in order to transfer knowledge between different domains with no need of so-called co-occurrence data. In their method, meta-features are defined to connect the features of the target and source spaces. In particular, the method learnt a mapping from each dimension in the target space to a corresponding dimension in the source space and then built a classifier on the labeled source data set along with the mapped labeled target data set if any for predicting the instances of the target domain. Different from [10], our method built a cluster model in the target space, directly on the target data set. Above all, both target and source data sets in our works are unlabeled. Zhou et al. (2015) [22] used a labeled source data set to train a classification system that could classify the instances in a different target data set by linking heterogeneous features with pivot features via joint non-negative matrix factorization. However, the method in [22] was specific for sentiment classification so that the authors could view the document instances in the source and target domains in the form of matrix. Besides, the authors constructed a common space and then built a prediction model in that space. Different from [22], our work aimed at enhancing the target data space and then built a clustering model in the enhanced target feature space.

Exceptionally among them are [3, 18] with the transfer learning process on unlabeled data in both source and target domains. Blitzer et al. (2006) [3] introduced structural correspondence learning to build a common space where a classifier learnt from the labeled source data set can be used for predicting the instances in the unlabeled target data set. In the transfer learning process, the authors examined the correlations between the domain-specific features via the pivot features with the pivot predictors. Pan et al. (2010) [18] proposed spectral feature alignment from spectral clustering on the unlabeled parts of the target and source data sets as we discussed previously in section 3. Compared to [18], the approach in [3] gave us more difficulties such as building a lot of pivot predictors and determining the labels for the data to build these predictors while our data sets are unlabeled. Therefore, our work was based on spectral feature alignment in [18] instead of structural correspondence learning in [3].

Secondly, many works such as [4, 5, 11-13] have been proposed for educational data clustering. Jovanovic et al. (2012) [12] performed the *k*-means clustering algorithm on the student's data related to cognitive styles and the score achieved for each course. A cluster model was built for each course to discover groups of the students with similar cognitive properties for e-learning improvements. Also with the *k*-means clustering algorithm, Campagni et al. (2014) [5] constructed the cluster models on the grades and delays of students for their examinations. The resulting clusters are then analyzed for highlighting the regularities over the years and used for course and student success improvement. Besides, Bresfelean et al. (2008) [4] used the *k*-means clustering algorithm with the FarthestFirst method to group the students. The resulting groups of similar students are utilized for building the two student's exam success and failure profiles. Different from [4, 5, 12], Jayabal and Ramanathan (2014) [11] proposed the Partitional Segmentation algorithm based on the PLS-path modeling approach to cluster the 10th grade data for analyzing the relationships of the performance in main subjects with medium of study. Moreover, Kerr and Chung (2012) [13] used the FANNY algorithm for fuzzy clustering and the AGNES algorithm for hard clustering in identifying key features of student performance from their different actions. As compared to these existing related works, our work focuses on a student clustering task based on the students' study performance at the program level with the assumption that our target data set has been collected in a small size for the target program. Therefore, we define another solution to this clustering task with a great support of another much larger data set gathered with another program. The situation leads to the inclusion of transfer learning in our proposed method as previously discussed.

Nevertheless, some recent works in the educational data mining area have also considered using multiple data sources in a mining task. They are listed as [14, 20]. In particular, Koprinska et al. (2015) [14] performed a student classification task with three data sources while Voβ et al. (2015) [20] conducted a performance prediction process with two different comparable data sets. Only Voβ et al. (2015) [20] suggested a transfer learning approach to reduce data sparseness with Matrix Factorization. In comparison with [20], our work has the different purpose, task, and approach for the transfer learning process. Indeed, transfer learning in our work is devoted to a target data space enhancement with spectral clustering on both target and source data sets so that the clustering process can then be executed more effectively on the target data set.

VI. CONCLUSION

In this paper, we have defined an educational data clustering task to cluster the undergraduate students into similar groups based on their study performance at the program level. In contrast to the educational data clustering tasks that have been solved in the existing works, our task is considered for the supported program A with which a small data set has been collected. Meanwhile, another program B has been supported and associated with a much larger data set for the task. Therefore, a solution to the task on the target data set of the program A is proposed in considering the exploitation of the source data set of the program B.

In particular, we have proposed a two-phase educational data clustering method based on transfer learning and kernel *k*-means algorithms as a solution to the task. This method has carried out the transfer learning process at the representation level in the first phase. It has concentrated on deriving the new features from the connections between the domain-specific features of the target and source data spaces with the domain-independent features shared by both target and source data spaces. These new features increase the capability of discriminating the instances of the target domain and are then used to enhance the target data space. In the second phase, our method has performed the clustering process at the instance level by means of kernel *k*-means. Different from the existing works on educational data clustering, the clustering process takes place in the enhanced target feature space instead of the original target data space or the enhanced target data space. Thus, the clusters can be formed naturally in arbitrary shapes. As a result from the experiments on real data sets, our cluster model is better than those from other approaches. The resulting clusters have been evaluated for their compactness and separation via the smaller objective function value, the smaller S_Dbw value, and the larger Dunn value for internal validation and the smaller Entropy value for external validation. Statistical tests also confirmed the significant better differences between our cluster models and the others.

As our future works, using the resulting cluster models in the educational decision support system is importantly considered for improving our students' study performance. They can be combined with a case based reasoning model for decision making support for academic affairs. They are also planned for generating study profiles of our students once their groups are observed. At that moment, their study trends can be kept track of towards their graduation.

ACKNOWLEDGMENT

This research is funded by Vietnam National University Ho Chi Minh City, Vietnam, under grant number C2016-20-16. Many sincere thanks also go to Mr. Nguyen Duy Hoang, M.Eng., for his support of the transfer learning algorithms in Matlab.

REFERENCES

1. AAO, Academic Affairs Office, www.aao.hcmut.edu.vn, accessed on 01/05/2017.
2. M. Belkin, P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (6) (2003) 1373-1396.
3. J. Blitzer, R. McDonald, F. Pereira. Domain adaptation with structural correspondence learning, Proc. The 2006 Conference on Empirical Methods in Natural Language Processing, 2006, pp. 120-128.
4. V. P. Bresfelean, M. Bresfelean, N. Ghisoiu. Determining students' academic failure profile founded on data mining methods, Proc. The ITI 2008 30th Int. Conf. on Information Technology Interfaces, 2008, pp. 317-322.
5. R. Campagni, D. Merlini, M.C. Verri. Finding regularities in courses evaluation with k-means clustering, Proc. The 6th Int. Conf. on Computer Supported Education, 2014, pp. 26-33.
6. W-C. Chang, Y. Wu, H. Liu, Y. Yang. Cross-domain kernel induction for transfer learning, AAAI (2017) 1-7.
7. F.R.K. Chung. Spectral graph theory, CBMS Regional Conference Series in Mathematics, No. 92, American Mathematical Society, 1997.
8. W. Dai, Q. Yang, G-R. Xue, Y. Yu. Self-taught clustering, Proc. The 25th International Conference on Machine Learning, 2008, pp. 1-8.
9. L. Duan, D. Xu, I. W. Tsang. Learning with augmented features for heterogeneous domain adaptation, Proc. The 29th International Conference on Machine Learning, 2012, pp. 1-8.
10. K. D. Feuz, D. J. Cook. Transfer learning across feature-rich heterogeneous feature spaces via feature-space remapping (FSR), ACM Trans. Intell. Syst. Technol. **6** (March 2015) 1–27.
11. Y. Jayabal, C. Ramanathan. Clustering students based on student's performance – a Partial Least Squares Path Modeling (PLS-PM) study, Proc. MLDM, LNAI 8556, 2014, pp. 393-407.

12. M. Jovanovic, M. Vukicevic, M. Milovanovic, M. Minovic. Using data mining on student behavior and cognitive style data for improving e-learning systems: a case study, International Journal of Computational Intelligence Systems **5** (2012) 597-610.

13. D. Kerr, G. K.W.K. Chung. Identifying key features of student performance in educational video games and simulations through cluster analysis, Journal of Educational Data Mining **4** (1)(October 2012) 144-182.

14. I. Koprinska, J. Stretton, K. Yacef. Predicting student performance from multiple data sources, Proc. AIED, 2015, pp. 1–4.

15. Y. Liu, Z. Li, H. Xiong, X. Gao, J. Wu. Understanding of internal clustering validation measures, Proc. the 2010 IEEE International Conference on Data Mining, 2010, pp. 911-916.

16. T. Martín-Wanton, J. Gonzalo, E. Amigó. An unsupervised transfer learning approach to discover topics for online reputation management, Proc. CIKM, 2013, pp. 1565-1568.

17. A. Y. Ng, M. I. Jordan, Y. Weiss. On spectral clustering: analysis and an algorithm, Advances in Neural Information Processing Systems 14, 2002, pp. 1-8.

18. S. J. Pan, X. Ni, J-T. Sun, Q. Yang, Z. Chen. Cross-domain sentiment classification via spectral feature alignment, Proc. WWW 2010, 2010, pp. 1-10.

19. G. Tzortzis, A. Likas. The global kernel k-means clustering algorithm, Proc. The 2008 International Joint Conference on Neural Networks, 2008, pp. 1978-1985.

20. L. Voβ, C. Schatten, C. Mazziotti, L. Schmidt-Thieme. A transfer learning approach for applying matrix factorization to small ITS datasets, Proc. The 8th International Conference on Educational Data Mining, 2015, pp. 372–375.

21. J. Wu, H. Xiong, J. Chen. Adapting the right measures for k-means clustering, Proc. KDD, 2009, pp. 877-885.

22. G. Zhou, T. He, W. Wu, X. T. Hu. Linking heterogeneous input features with pivots for domain adaptation, Proc. The 24th International Joint Conference on Artificial Intelligence, 2015. pp. 1419-1425.

PHƯƠNG PHÁP GOM CỤM DỮ LIỆU GIÁO DỤC HAI GIAI ĐOẠN DỰA TRÊN HỌC CHUYỂN ĐỔI VÀ KERNEL K-MEANS

Tóm tắt: Trong bài báo này, chúng tôi đề xuất phương pháp gom cụm dữ liệu giáo dục hai giai đoạn dựa trên học chuyển đổi và kernel k-means. Phương pháp này là giải pháp cho bài toán gom cụm sinh viên với tập dữ liệu đích được thu thập từ chương trình đích là ít; trong khi đó, chương trình nguồn khác lại đang có sẵn tập dữ liệu lớn hơn nhiều. Tập dữ liệu đích ít trong không gian dữ liệu cao chiều có thể không đủ tốt cho bài toán gom cụm này. Do đó, phương pháp được đề xuất quyết định triển khai học chuyển đổi ở giai đoạn đầu để khai thác cả hai tập dữ liệu nguồn và đích không có nhãn nhằm phát triển dạng biểu diễn tốt hơn cho các đối tượng trong miền đích. Một số đặc trưng mới được dẫn ra ở giai đoạn đầu này với gom cụm phổ trên các đặc trưng độc lập miền và các đặc trưng phụ thuộc miền của cả hai miền đích và nguồn. Các đặc trưng mới này sẽ được dùng để tăng cường không gian dữ liệu của miền đích. Trong giai đoạn sau, phương pháp được đề nghị sẽ thực thi giải thuật kernel k-means để hình thành cụm của các sinh viên trong không gian đặc trưng được tăng cường của miền đích. Thật ra các cụm này trở thành các cụm có hình dạng tùy ý trong không gian dữ liệu được tăng cường của miền đích với độ nén và độ phân tách tốt hơn. So với các công trình liên quan hiện có trong lĩnh vực khai phá dữ liệu giáo dục, bài toán và phương pháp tương ứng được đề xuất là mới cho việc gom các sinh viên tương tự nhau vào các nhóm tương ứng dựa trên kết quả học tập ở mức chương trình của các sinh viên này. Hơn nữa, các kết quả thực nghiệm và kiểm định thống kê trên các tập dữ liệu thực tế đã cho thấy tính hiệu quả của phương pháp được đề xuất với các cụm có chất lượng tốt hơn so với các cụm có được từ các hướng tiếp cận khác.

Từ khóa: Gom cụm dữ liệu giáo dục, kernel k-means, học chuyển đổi, thích nghi miền theo hướng tiếp cận không giám sát, khoảng cách Euclidean trong không gian biến đổi dựa trên kernel.

Vo Thi Ngoc Chau, currently with Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam National University-HCMC, Vietnam. She got a bachelor’s degree in Information Technology from Ho Chi Minh City University of Technology, Vietnam National University-HCMC, Vietnam, in 2003, a master’s degree and a doctoral degree from King Mongkut’s Institute of Technology Ladkrabang, Thailand, in 2005 and 2008, respectively, under the AUN/Seed-Net scholarships from JICA. Her research of interest includes complex data modeling and analysis, data mining and knowledge discovery, decision support systems and other modern information systems.

Nguyen Hua Phung, a senior lecturer in the Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam National University-HCMC, Vietnam. He received his Ph.D. (2005) in Computer Science from UNSW, Australia, and M.Eng (1999) and B.Eng in Computer Engineering from Ho Chi Minh City University Of Technology, Vietnam National University-HCMC, Vietnam. His research interests include scheduling, data mining, program analysis and verification.