Clustering of online learning resources via minimum spanning tree

Qingyuan Wu and Changchen Zhan
School of Management, Beijing Normal University, Zhuhai, China
Fu Lee Wang
Caritas Institute of Higher Education, Hong Kong, Hong Kong
Siyang Wang
Sun Yat-sen University, Guangzhou, China, and
Zeping Tang
Huawei Technologies Co., Ltd, Shenzhen, China

Abstract

Purpose – The quick growth of web-based and mobile e-learning applications such as massive open online courses have created a large volume of online learning resources. Confronting such a large amount of learning data, it is important to develop effective clustering approaches for user group modeling and intelligent tutoring. The paper aims to discuss these issues.

Design/methodology/approach – In this paper, a minimum spanning tree based approach is proposed for clustering of online learning resources. The novel clustering approach has two main stages, namely, elimination stage and construction stage. During the elimination stage, the Euclidean distance is adopted as a metrics formula to measure density of learning resources. Resources with quite low densities are identified as outliers and therefore removed. During the construction stage, a minimum spanning tree is built by initializing the centroids according to the degree of freedom of the resources. Online learning resources are subsequently partitioned into clusters by exploiting the structure of minimum spanning tree.

Findings – Conventional clustering algorithms have a number of shortcomings such that they cannot handle online learning resources effectively. On the one hand, extant partitional clustering methods use a randomly assigned centroid for each cluster, which usually cause the problem of ineffective clustering results. On the other hand, classical density-based clustering methods are very computationally expensive and time-consuming. Experimental results indicate that the algorithm proposed outperforms the traditional clustering algorithms for online learning resources.

Originality/value – The effectiveness of the proposed algorithms has been validated by using several data sets. Moreover, the proposed clustering algorithm has great potential in e-learning applications. It has been demonstrated how the novel technique can be integrated in various e-learning systems. For example, the clustering technique can classify learners into groups so that homogeneous grouping can improve the effectiveness of learning. Moreover, clustering of

The work described in this paper was supported by a grant from the Soft Science Research Project of Guangdong Province (Grant No. 2014A030304013), and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (UGC/FDS11/E06/14).

Asian Association of Open Universities Journal
Vol. 11 No. 2, 2016
pp. 197-215
Emerald Group Publishing Limited
1858-3431
DOI 10.1108/AAOUJ-09-2016-0036

online learning resources is valuable to decision making in terms of tutorial strategies and instructional design for intelligent tutoring. Lastly, a number of directions for future research have been identified in the study.

Keywords Clustering, E-learning, Density based, Minimum spanning tree, Online learning resources
Paper type Research paper

1. Introduction

E-learning is a means of education that incorporates self-motivation, communication, efficiency, and technology (Phobun and Vicheanpanya, 2010; Woldab, 2014). As a general tendency in intelligent tutoring and learning, e-learning has attracted an increasing amount of attention from researchers in the fields of computer science, pedagogy, and praxeology. With the rapid growth of e-learning resources, including content delivered through the internet, intranet/extranet, CD-ROM, audio or video tape, and satellite TV, the selection and organization of these materials is very time-consuming and challenging to users. Thus, it is necessary to cluster learning resources and subsequently recommend personalized resources to both teachers and learners. Clustering is the process of assigning class labels to objects based on the principle of minimizing the interclass similarity and maximizing the intraclass similarity (Li *et al.*, 2013), which is widely used in various scientific areas (Ben *et al.*, 2011). For instance, taxonomists, social scientists, psychologists, biologists, statisticians, mathematicians, engineers, computer scientists, medical researchers, and others who collect and process real-world data have all contributed to clustering methodology (Jain, 2010; Mimaroglu and Erdil, 2011). At the same time, a recent trend in e-learning is the development of massive open online courses (MOOCs) and micro-courses. With the help of numerous teachers, MOOCs provide unlimited participation and open access via the internet to learners worldwide. It is not rare that tens of thousands of students from around the world enroll in a single course. As more and more learning resources are generated due to the explosion of MOOCs, it is hard to apply traditional clustering algorithms to analyze online learning resources.

Outliers or noise objects are very common in real-world data sets, especially for user-generated content. This brings new challenges to existing clustering methods. On the one hand, most of traditional partitional clustering algorithms (e.g. *K*-means, bisecting *K*-means, and *K*-medoids) randomly assign objects as initial centroids of the clusters. Outliers may be chosen as the initial centroids of clusters. It will then converge to an unstable result (i.e. instability issue). On the other hand, the performance of classical density-based clustering methods (e.g. density-based spatial clustering of applications with noise (DBSCAN)) will be computationally expensive and time-consuming when they are facing those noise objects.

In this paper, a novel scheme is proposed to resolve the problem of instability and inefficiency for clustering of online learning resources. Outliers are first eliminated based on the density of each resource. Then, a minimum spanning tree is constructed based on the distances among resources. The degree of freedom for each resource is subsequently calculated based on the structure of the minimum spanning tree. The resource with the largest value of degree of freedom will be considered as the initial centroid. In comparison with the previous work (Wang *et al.*, 2015), a number of enhancements have been made: a more comprehensive literature review has been conducted in Section 2; the effectiveness of the proposed algorithm together with other clustering algorithms are evaluated with two additional data sets including a two-dimensional data set (Section 4.3) and a real-world e-learning data set (Section 4.4) to improve the generalization of results; one classical density-based clustering method (i.e. DBSCAN) is implemented for comparison, and the experimental results are

analyzed in more detail; and more detailed information and in-depth discussion is provided in the introduction, experiments, conclusion, and future research directions.

The rest of the paper is organized as follows. Section 2 describes related work on e-learning systems and clustering of online learning resources. Section 3 presents a novel clustering algorithm based on minimum spanning tree. Section 4 evaluates clustering algorithms with four data sets. Section 5 discusses the directions of incorporating the proposed clustering algorithm into e-learning systems. Finally, Section 6 provides concluding remarks.

2. Related works

2.1 E-learning systems

E-learning is valuable to educational institutions, corporations and all types of learners as it eliminates distances and subsequent commutes (Phobun and Vicheanpanya, 2010). It is affordable and time-saving because a wide range of online learning resources can be accessed from properly equipped computer terminals. Thus, the development of e-learning systems is one of the fastest growing trends in educational uses of technology (Li *et al.*, 2009). Applications and components of e-learning systems include construction of learning models, prediction of learners' learning behavior, development of mobile application, and so forth. For instance, Zou *et al.* (2014) proposed an incidental word learning model for e-learning. In particular, they measured the load of various incidental word learning tasks so as to construct load-based learner profiles. A task generation method was further developed based on the learner profile to increase the effectiveness of various word learning activities. Boyer and Veeramachaneni (2015) designed a set of processes which take the advantage of knowledge from both previous courses and previous weeks of the same course to make real-time prediction on learners' behavior. Ferschke *et al.* (2015) implemented a Lobby program that students can be connected via a live link at any time. Zbick (2013) presented a web-based approach to provide an authoring tool for creation of mobile applications with data collection purposes.

2.2 Clustering of learning resources

It is believed that e-learning systems should provide a variety of learning resources to satisfy need of different learners (Sabitha *et al.*, 2016). With the rapid growth of online learning resources, learners are facing a serious problem of information overload. A tool is urgently required to assist the learners to get the similar learning materials efficiently. The clustering algorithms are extensively employed for discovery of community (Xie *et al.*, 2012, 2014) and event detection (Rao and Li, 2012), which are important research topics in e-learning. Sabitha *et al.* (2016) employed fuzzy clustering technique to combine learning and knowledge resources based on attributes of metadata. Mansur and Yusof (2013) tried to reveal the behavior of students from all activities in Moodle e-learning system by using ontology clustering techniques. In their ontology model, the forum, quiz, assignment, and many other activities were placed as clustering parameters. Govindarajan *et al.* (2013) employed particle swarm optimization algorithm to analyze and cluster continuously captured data from students' learning interactions. However, some useless resources may exist in e-learning systems.

It is important to remove the noise objects before clustering. Mimaroglu and Erdil (2011) defined two variables named weight and attachment to address the issue of noise object. The first one (i.e. weight) measures the similarity between two objects, and the second one (i.e. attachment) ranks the quality of each candidate centroid. Noise objects are removed based on their measurement of weight and attachment. Luo *et al.* (2010) proposed another

method to exclude the "fake" centroid based on the notion of density, as follows: let $X = \{x_1, x_2, \ldots, x_n\}$ be the set of objects. $DEN(x_i)$ is the density of object x_i. A small value of $DEN(x_i)$ indicates that x_i locates at a relative high-density location, or vice versa. The density of x_i is compared with the average density $ADEN$. If an object has density higher than the average density, it will be considered as a "fake" centroid and therefore eliminated.

3. Clustering of online learning resources

3.1 The overall framework

The increasing availability of digital educational materials on the internet, called online learning resources, has been followed by the definition of indexing standards. However, the selection process of these elements is challenging to learners because of the diversity of metadata approaches, in addition to the lack of consensus about the definition of learning resources (Silva and Mustaro, 2009). In light of these considerations, learners need effective and efficient clustering methods to organize and manage such large volume of online learning resources. The objective of clustering of online learning resources in this study is to assign class labels to various learning resources by eliminating outliers, and to improve the accuracy of clustering algorithm based on the minimum spanning tree as well as procedures of merging learning resources and small clusters.

As illustrated in Figure 1, a clustering framework for online learning resources with four key steps is proposed as follows:

(1) The density of each instance of online learning resource is measured in order to identify and eliminate outliers. Learning resources that are few and scattered in their areas will be removed in this step.

(2) A minimum spanning tree is constructed to create a link of all learning resources. The minimum spanning tree is helpful to detect clusters of different shapes and sizes (Päivinen, 2005).

(3) A partitioning method based on the structure of minimum spanning tree is employed to merge learning resources into clusters.

(4) The small clusters that contain only a few learning resources are also merged into large ones.

The density-based clustering algorithm proposed in this paper can be applied to a number of areas in e-learning, for example, classification of learner, discovery of learning path, recommendation of learning resource, and intelligent tutoring.

On the other hand, the key parameters of the proposed approach are explained below in the context of online learning resources for better understanding of the paper:

- distance between two online learning resources measures the dissimilarity between the contents of two resources;
- density of an online learning resource measures number of learning resources which are similar as the resource, i.e., their distances to the learning resource are less than a threshold value;
- outlier or noise learning resource is a learning resource which is very different from the others; and
- usefulness of learning resource refers to the relevancy of the learning resource to the learner's study or learning interest.

Mathematical definitions of the parameters can be found in the following subsection.

3.2 Elimination of outliers

The existence of outliers will produce useless learning resources in e-learning, and disturb the effect of clustering. In order to solve this problem, the method proposed by Luo *et al.* (2010) is incorporated in the algorithm proposed. The related definitions are shown below:

Definition 1. The density of an object (i.e. an online learning resource) is:

$$DEN(x_i) = \frac{1}{m} \sum_{y_j \in \varphi(x_i)} d(x_i, y_j),$$

where $\varphi(x_i)$ is the set of m nearest online learning resources of x_i, $d(x_i,y_i)$ is the Euclidean distance (Deza and Deza, 2016) between x_i and y_j.

Definition 2. The average density of online learning resources is:

$$ADEN = \frac{1}{p} \sum_{i=1}^{p} DEN(x_i),$$

where p is the number of online learning resources.

Lemma 1. The density *DEN* of some normal online learning resources is larger than the average density *ADEN*.

Proof 1: if all online learning resources are normal, i.e., no outliers, the value of *ADEN* must between DEN_{max} and DEN_{min}. Thus, the density *DEN* of some normal online learning resources must be larger than then average distance *ADEN*. ■

According to Lemma 1, a constant *DEV* is added to the average distance *ADEN*. If $DEN(x_i)$ is larger than the sum of *ADEN* and *DEV*, it will be considered as an outlier and removed from the data set.

3.3 Generation of minimum spanning tree

After the elimination of noise resources, there are still a huge number of online learning resources. As a result, an efficient clustering technique is required to group similar learning resources together as clusters. The distance between each pair of remaining objects is first calculated, and then the minimum spanning tree of remaining learning resources is built accordingly by using the Prim's (1957) algorithm. A minimum spanning tree is constructed to create a link of all remaining objects (Algorithm 1).

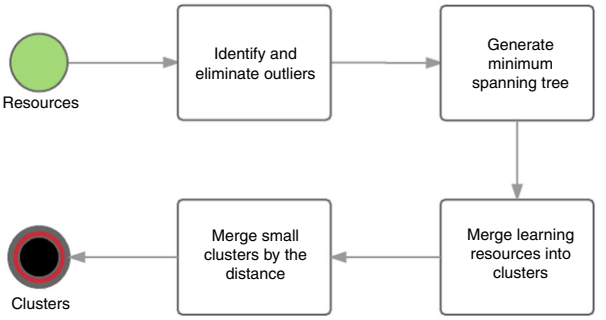

Figure 1. Framework of online learning resource clustering

Algorithm 1. Algorithm of generating the minimum spanning tree.
Input: A weighted connected graph, with a vertex set V and an edge set E;
Output: A set V_{new} and a set E_{new} by which the minimum spanning tree is described
1: initialization: $V_{new} = \{x\}$ (x is the starting point chosen from V), E_{new} = empty;
2: while $V_{new} \neq V$ do
3: choose edge $< u; v >$ with minimum weight from $E (u \in V_{new}, v \notin V_{new}$ and $v \in V)$;
4: add v into V_{new} and add $< u; v >$ into E_{new};
5: end while

3.4 Merging learning resources into clusters

In the previous subsection, a minimum spanning tree is generated. The degree of freedom of each instance of learning resources can be obtained by using Definition 3:

Definition 3. The degree of freedom of an object (i.e. online learning resource) x_i is:

$$df(x_i) = \left|\left\{x_j \middle| (x_i, x_j) \in E\right\}\right|,$$

where E denotes the edges that x_i belongs to.

It is believed that an object with a large value of degree of freedom means that it has a large number of neighbors. The object therefore may be a centroid (Mimaroglu and Erdil, 2011). Thus the learning resources are sorted according to their degree of freedom. Subsequently, the learning resources are partitioned into clusters based on the structure of minimum spanning tree and their degree of freedom (Algorithm 2).

Figure 2 provides an example to demonstrate the operation of Algorithm 2. Table I shows the Euclidean distance between all pairs of objects.

This algorithm is illustrated as follows:

(1) Six objects $v_1, v_2, \ldots, v_6$ are used as an example (Figure 2). The parameters *DEV* and m are set as 0.5 and 2, respectively. The values of density *DEN* of each object are shown in Table II. Because there is no noise object, all objects are hence reserved.

(2) A minimum spanning tree for the objects is generated by using the Prim's algorithm. The resulting edges of the tree are (v_1, v_2), (v_2, v_3), (v_1, v_4), (v_4, v_5), (v_5, v_6).

(3) Table II shows the degree of freedom of each object. The objects are sorted in reverse order of their degree of freedom. The order of the objects after sorting is $v_1, v_2, v_4, v_5, v_3, v_6$. Thus, object v_1 is put into the first cluster, i.e., Cluster 1.

(4) The immediate neighboring objects of v_1 are v_2 and v_4. Object v_2 is put into Cluster 1 because $d(v_1, v_2) < d(v_2, v_3)$. Object v_4 is not included in Cluster 1 because $d(v_1, v_4) > d(v_4, v_5)$.

(5) The neighboring objects of the newly added object are subsequently considered. Because object v_2 is the newly added object, object v_3 which is the immediate neighbor of object v_2 is considered. Object v_3 is put into Cluster 1 because v_3 have no other neighboring objects and the minimum distance between object v_3 and its neighboring objects is $d(v_2, v_3)$. Now, Cluster 1 has three objects, i.e., $\{v_1, v_2, v_3\}$.

(6) The object with the largest value of degree of freedom is first chosen among remaining objects. Among the three objects v_4, v_5, and v_6, object v_4 has the

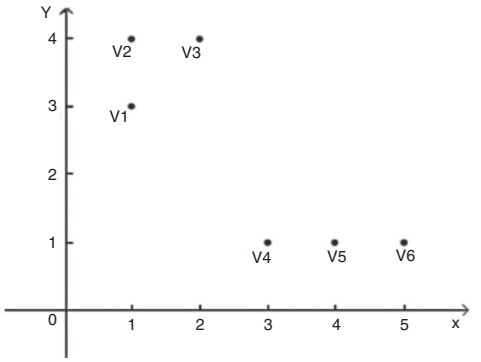

Figure 2. Data before clustering

	v_1	v_2	v_3	v_4	v_5	v_6
v_1	–	1.000	1.414	2.828	3.606	4.472
v_2	1.000	–	1.000	3.606	4.243	5.000
v_3	1.414	1.000	–	3.162	3.606	4.243
v_4	2.828	3.606	3.162	–	1.000	2.000
v_5	3.606	4.243	3.606	1.000	–	1.000
v_6	4.472	5.000	4.243	2.000	1.000	–

Table I. The Euclidean distance between objects

Object	v_1	v_2	v_3	v_4	v_5	v_6
DEN	1.207	1.000	1.207	1.500	1.000	1.500
df	2	2	1	2	2	1

Table II. The value of every object

largest value of degree of freedom. The neighboring objects of v_4 are v_1 and v_5. As object v_1 has been put into Cluster 1, it is not considered here. Object v_5 is put into Cluster 2 because $d(v_4, v_5) = d(v_5, v_6)$. After that, the neighboring objects of the object which is newly added are considered, i.e., object v_6. Object v_6 is also added into Cluster 2 because it has no other neighboring objects, and the minimum distance between object v_6 and its neighboring objects is $d(v_5, v_6)$. At last, all objects have been put into clusters and two clusters are generated by the algorithm, i.e., Cluster 1 = $\{v_1, v_2, v_3\}$ and Cluster 2 = $\{v_4, v_5, v_6\}$ (Figure 3).

Algorithm 2. A partitioning method based on minimum spanning tree.

Input: *D*: Date Set
Output: *C*: Clusters
1: Calculate the Euclidean distance between each pair of objects;
2: Calculate the Density *DEN* based on the Euclidean distance;
3: Calculate the average density *ADEN*;
4: for all objects do
5: if *DEN* > *ADEN* + *DEV* then
6: Elimination of outlier
7: end if
8: end for

```
Construct a minimum spanning tree for remaining objects by using the Prim's algorithm;
Calculate and sort the degree of freedom of each vertex in the tree;
cluster = 1;
while there are unmarked objects do
    Add unmarked object with the highest value of degree of freedom to an empty queue;
    while queue is not empty do
        V = dequeue();
        Add V to the present cluster;
        Mark V ;
        for all edge < v,w > do
            if w is unmarked then
                weight = weight(v,w);
            end if
            for all edge < w, t > do
                if weight(w, t)⩽ weight then
                    is_max = false;
                    break;
                end if
            end for
            if is_max = true then
                Enqueue(w);
            end if
        end for
    end while
    cluster++;
end while
for every cluster obtained above do
    if the number of objects in the cluster < minNum then
        if distance between the cluster and neighboring cluster < minDis then
            merge the cluster into the neighboring cluster;
        end if
    end if
end for
```

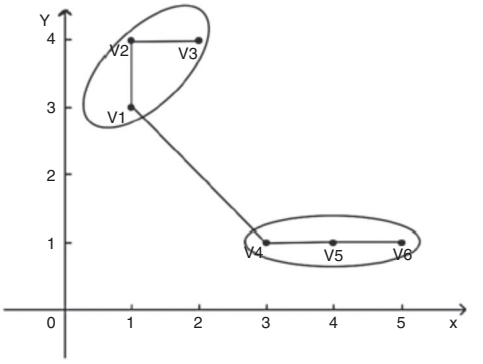

Figure 3.
Data after clustering

3.5 Merging small clusters by the distance

Based on the minimum spanning tree generated, an initial clustering result is obtained by using Algorithm 2. However, there may be a large number of small clusters which only contain a few learning resources. To save computational resources, the small clusters will be further merged into large clusters. Algorithm 3 details the merging of small clusters into large clusters, where *minNum* indicates the minimum number of objects required for a cluster, *minDis* represents the minimum distance between clusters. If the number of objects in one cluster is less than *minNum* and the distance between the cluster and its closest neighboring cluster is less than *minDis*, the cluster is merged into its closest neighboring.

Algorithm 3. Merging small clusters based on the distance.
1: for every cluster obtained above do
2: if the number of objects in cluster $<$ *minNum* then
3: if distance between the cluster and its closest neighboring cluster $<$ *minDis* then
4: merge the cluster into its closest neighboring cluster;
5: end if
6: end if
7: end for

3.6 Comparison with technique proposed with density-based clustering methods

Clustering approaches are very popular for understanding the natural grouping or structure in a data set. There are various clustering algorithms such as *K*-means, bisecting *K*-means, *K*-medoids, and fuzzy-means clustering. The main drawback of those approaches is the random selection of initial centroids (i.e. instability issue). In addition, the traditional clustering approaches can find only spherical-shaped clusters (Govindarajan *et al.*, 2013). Other clustering methods have been developed for non-spherical cluster shape based on the notion of density. Density-based clustering can be used to filter out noise objects (outliers) and discover clusters of arbitrary shape effectively (Duan *et al.*, 2006).

DBSCAN is one of the most widely used density-based clustering algorithms, which can discover clusters of arbitrary shape in spatial databases with noise objects (Ester *et al.*, 1996). The general idea of DBSCAN is that for each instance of a cluster, the neighborhood of a given radius (ε) has to contain at least a minimum number of points (*MinPts*), where ε and *MinPts* are parameters set by users manually. If a spatial index is used, the computational complexity of DBSCAN is $O(n\log n)$, where n is the number of objects. Otherwise, its computational complexity is $O(n^2)$.

In this paper, a novel clustering technique is proposed based minimum spanning tree. Because the minimum spanning tree is built by using the Prim's algorithm and the running time of Prim's algorithm is $O(n^2)$, the overall running time for the technique proposed is also $O(n^2)$. In this regard, the computational complexity of the technique proposed is comparable with DBSCAN. The efficiency of DBSCAN is highly dependent on appropriate settings of the user-defined parameters ε and *MinPts*. The performance of DBSCAN will be computationally expensive and time-consuming when they are facing noise objects. The proposed technique is free of the problem of noise objects, because they are removed at the early stage.

4. Experiments

In this section, the clustering technique proposed is evaluated by using four different data sets. First, we employ three data sets (i.e. "Smileface," "Aggregation," and "Jain"

data sets) to test the effectiveness of our method and standard clustering algorithms, because these data sets have quite different densities, scales, and shapes. Second, a large-scale discussion threads from the forums of Coursera MOOCs is used for real-world validation. Specifically, the "Smileface" data set contains clusters with both uniform and uneven densities, which is suitable to evaluate the effectiveness of density-based clustering algorithms. The "Aggregation" data set has seven clusters with different scales, and the "Jain" data set contains two clusters with ambiguous boundaries. The above features may be presented in online learning resources and will bring challenges to clustering approaches. The classical *K*-means clustering, average-link hierarchical clustering, complete-link hierarchical clustering, and DBSCAN method are implemented in this study for comparison.

4.1 Results of different clustering algorithms on the "Smileface" data set

The algorithm proposed is first evaluated with the data set named "Smileface." The "Smileface" data set contains a total of 644 points which belong to four different clusters.

The *K*-means clustering performs very well with data points in globular shaped clusters. However, clusters in the "Smileface" data set are not in the globular shape. Figure 4 shows the result of *K*-means clustering with *K* equals to 4. Figures 5 and 6 show the results of average-link hierarchical clustering algorithm and complete-link hierarchical clustering algorithm, respectively. It is observed that the four clusters are not separated very well by three baseline algorithms.

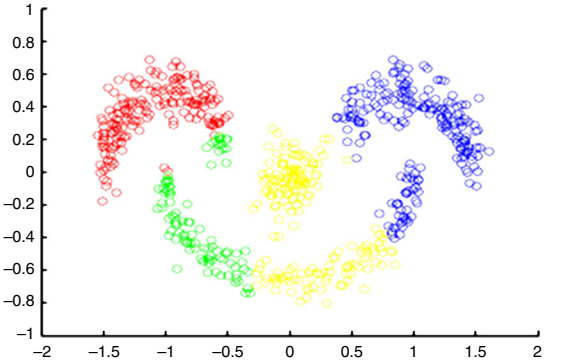

Figure 4.
Result of *K*-means on "Smileface" data set

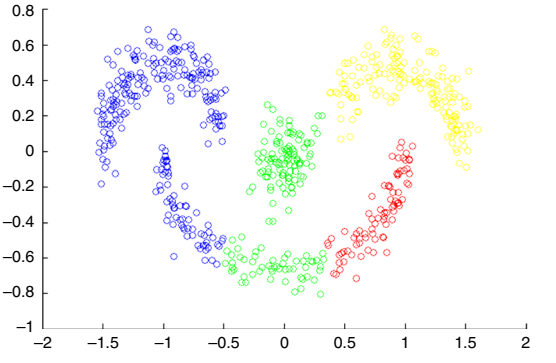

Figure 5.
Result of average-link on "Smileface" data set

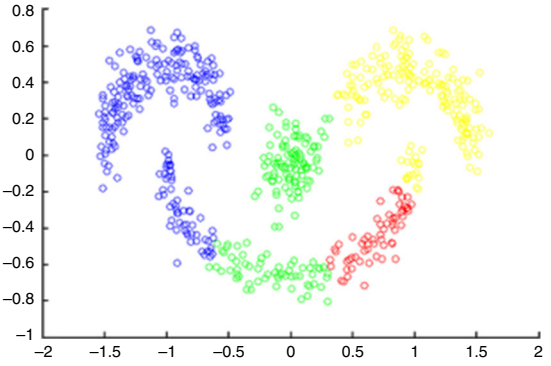

Figure 6. Result of complete-link on "Smileface" data set

Figure 7 shows the result of the proposed algorithm with parameters shown in Table III. As shown in Figure 7, our clustering scheme is robust and it can handle the outliers very well. The result of clusters produced by our algorithm is more satisfactory than three baseline algorithms.

4.2 Results of different clustering algorithms on the "aggregation" data set

Similarly, all clustering algorithms are also evaluated with the data set named "Aggregation." The "Aggregation" data set contains a total of 788 points, which belong to seven different clusters. In comparison with the "Smileface" data set, the "Aggregation" data set is more complex.

Figure 8 shows the result of *K*-means clustering. It is observed that the red cluster contains points which belong to three different clusters. Furthermore, two clusters in the right hand side with internal touch are separated into three clusters. Figures 9 and 10 show the results of average-link hierarchical clustering and complete-link hierarchical clustering algorithms, respectively. The result of the average-link

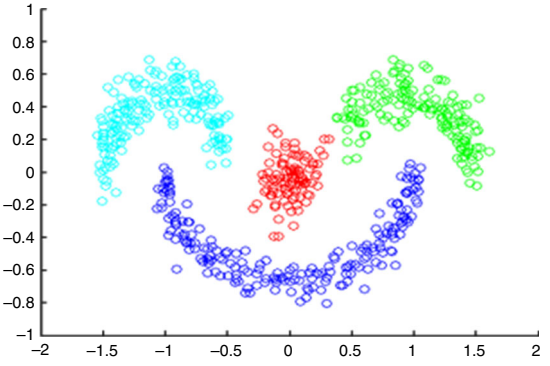

Figure 7. Result of our algorithm on "Smileface" data set

Point	*m*	*DEV*	*minNum*	*minDis*
Value1	10	0.5	80	0.5
Value2	30	0.5	70	2

Table III. The value of parameters

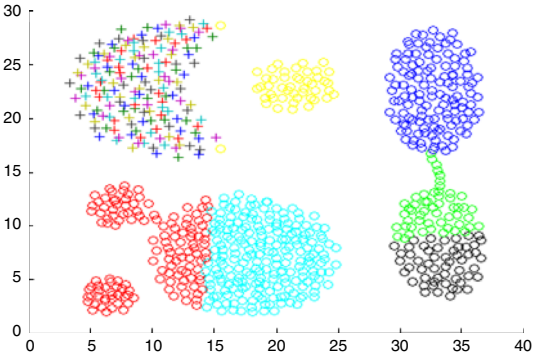

Figure 8.
Result of *K*-means on "Aggregation" data set

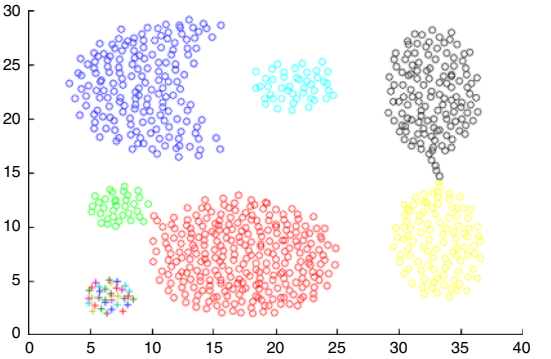

Figure 9.
Result of average-link on "Aggregation" data set

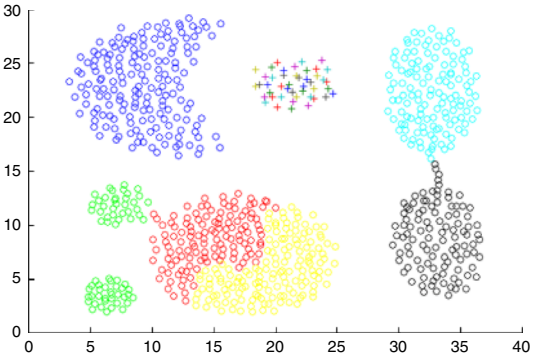

Figure 10.
Result of complete-link on "Aggregation" data set

hierarchical clustering is good. However, the complete-link hierarchical algorithm performs very poorly on the "Aggregation" data set.

Figure 11 shows the experimental result of our algorithm. It exactly separates two clusters in the right hand side which are wrongly separated by *K*-means clustering. However, it groups the points which belong to three clusters in the lower left corner into two clusters. This problem will be further investigated in the future. It provides a research direction for enhancement of our algorithm proposed.

4.3 Results of different clustering algorithms on the "Jain" data set

A two-dimensional data set "Jain" is used for further evaluation of the robustness of the clustering algorithms over clusters with different densities. The "Jain" data set contains a total of 373 points, which belong to two clusters. Different from the aforementioned two data sets, the cluster densities of this data set are different with each other.

Figure 12 shows the result of *K*-means clustering. It is observed that the performance of *K*-means on "Jain" data set is poor since these two clusters are not in the globular shape. Figures 13 and 14 show the results of average-link hierarchical

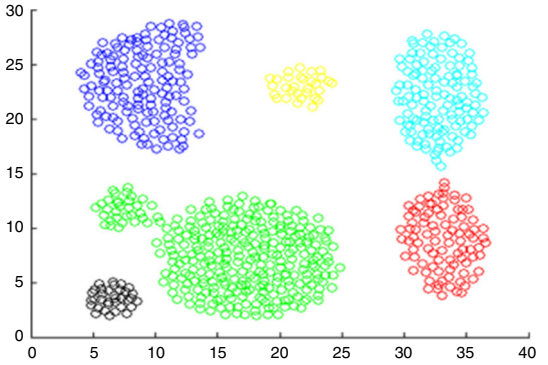

Figure 11.
Result of our algorithm on "Aggregation" data set

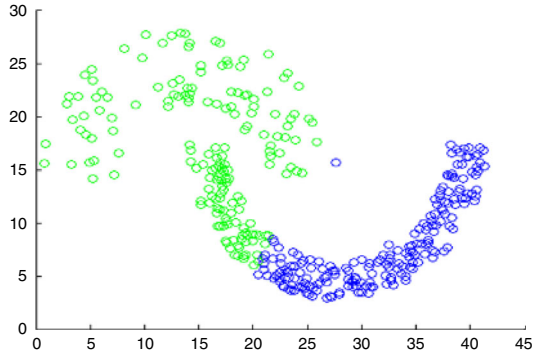

Figure 12.
Result of *K*-means on "Jain" data set

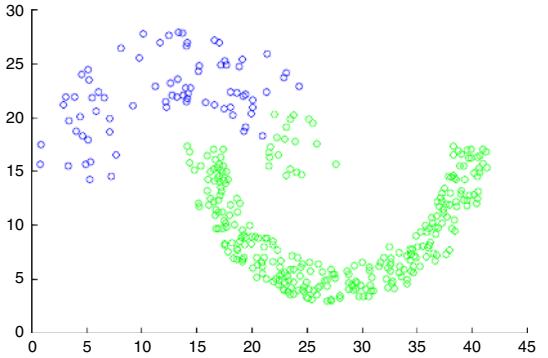

Figure 13.
Result of average-link on "Jain" data set

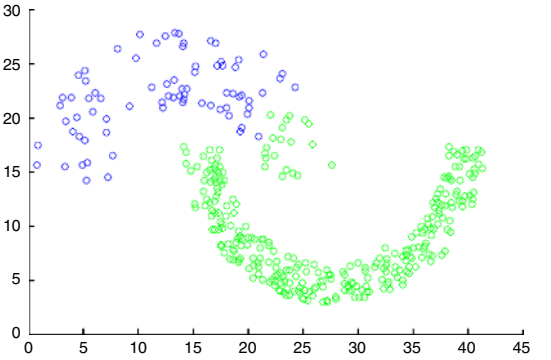

Figure 14.
Result of complete-link on "Jain" data set

clustering and complete-link clustering algorithms, respectively. In this case, it is observed that these two algorithms have the same experimental results. Figure 15 shows the result of our algorithm which treats the points with low density as noise points and eliminates them. This indicates that our algorithm is more suitable to identify dense resources than other baseline methods.

4.4 Results of different clustering algorithms on the e-learning data set

In this section, the proposed minimum spanning tree based clustering algorithm is compared with the classical density-based algorithm DBSCAN, and the best-performing baseline of average-link hierarchical clustering by using a real-world e-learning data set (Rossi and Gnawali, 2014). This data set is the anonymized version of the discussion threads from the forums of 60 Coursera MOOCs, for a total of about 100,000 threads. After removing the redundant items, 73,942 learning instances are used for evaluation. A total of 197 distinct courses are assigned to four clusters (i.e. automata-002, bigdata-edu-001, humankind-001, and gametheory-003).

The characteristics of the "MOOCs" data set are used as the density of DBSCAN, and the density of the object is used as the parameter of our minimum spanning tree algorithm.

The actual clusters of the MOOCs data set are shown in Figure 16. By tuning various combination of parameters, the best clustering result of DBSCAN is shown in Figure 17. The result of average-link hierarchical clustering, which performed well on the previous "Aggregation" data set, is shown in Figure 18. However, it is observed

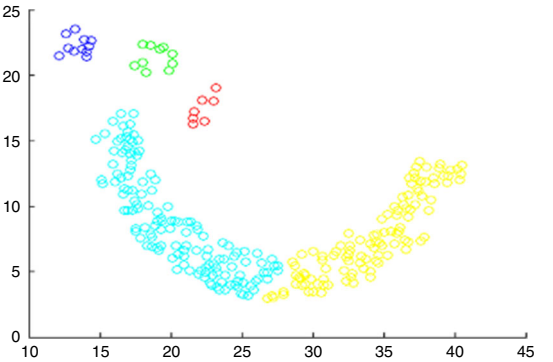

Figure 15.
Result of our algorithm on "Jain" data set

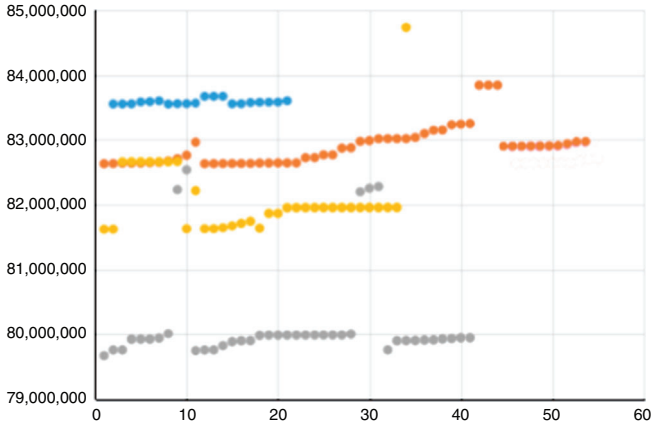

Figure 16.
Actual clusters of "MOOCs" data set

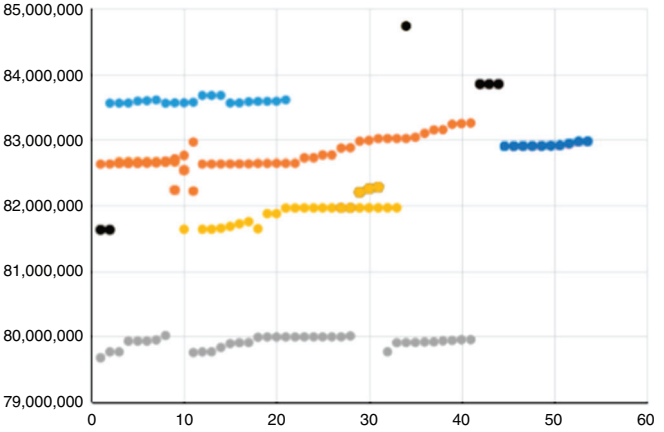

Figure 17.
Result of DBSCAN on "MOOCs" data set

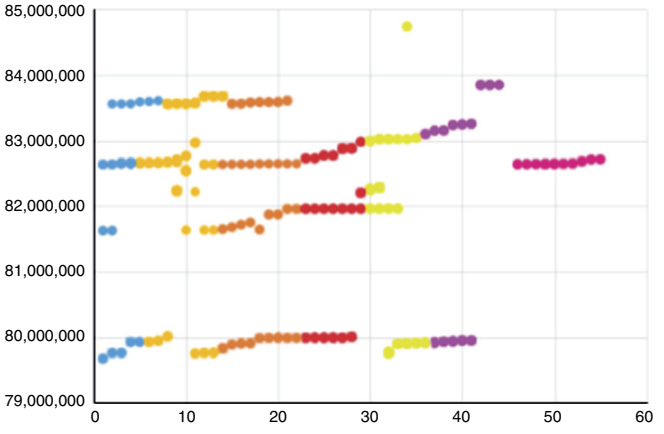

Figure 18.
Result of average-link on "MOOCs"data set

that these two baselines both generated some errors on the e-learning data set. The clustering result of the algorithm proposed is shown in Figure 19, which is nearly the same with the grand truth (Figure 16).

On the one hand, the proposed minimum spanning tree based clustering algorithm shows higher accuracy, which can group the online courses into clusters effectively. On the other hand, our algorithm can find the appropriate parameters efficiently on the "MOOCs" data set, i.e., it is robust to make a correct distinction between the labeled and unlabeled e-learning data sets.

The experimental results also indicate that determination of parameters for different clustering algorithms on e-learning data sets is a critical factor which affects the effectiveness of the algorithms. This provides another direction for future research.

5. E-learning applications

This section will discuss briefly how to apply the novel clustering technique proposed in various e-learning systems. Generally, the algorithm proposed can be employed in the following four aspects.

5.1 Classification of learner

As there may be more than tens of thousands of learners enrolling in one single course in MOOCs, it is very important to cluster the learners into groups. The effectiveness of learning can be greatly improved by homogenous grouping. Because the learners in a group have common characteristics, so that learning material and teaching strategies can be adjusted accordingly. For instance, the MITx and HarvardX (2014) data set suggests that the learners with good academic results have similar pattern of playing of course video. These learners are quite close with each other if their attributes (e.g. frequency of playing video) are plotted in an n-dimensional graph. As a result, the clustering algorithm proposed can differentiate the learners and corresponding assistances can be offered subsequently.

5.2 Discovery of learning path

Discovery of learning path is a classical application in e-learning system. On the other hand, it is very time-consuming and extremely challenging for users to identify

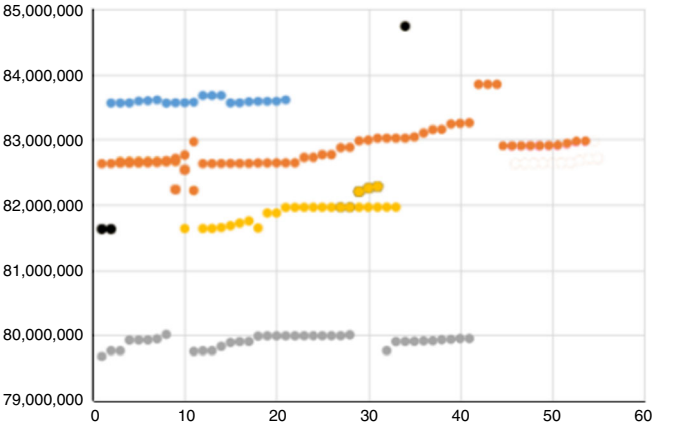

Figure 19.
Result of our algorithm on "MOOCs" data set

their optimized learning paths when they wish to acquire new knowledge in a specific topic. A key step in discovery of learning path is to identify whether there is a strong linkage between two knowledge units (Leung and Li, 2003). It can be easily determined whether two knowledge units are in the same cluster by using the clustering result produced by the method proposed. It is less time-consuming, because it is not required to compare all pairs of knowledge units.

5.3 Recommendation of learning resource

In web-based learning, learners are facing a problem of overloading of online learning resources. It is essential to identify suitable learning resources from a potentially overwhelming variety of choices (Manouselis *et al.*, 2010). The algorithm proposed can discover the natural grouping of online learning resources effectively. The system can easily recommend both interesting and relevant learning resources to learners by using the clustering result.

5.4 Intelligent tutoring

Intelligent tutoring is a generation of learning oriented methodology that includes the individuality of the learner in the learning process. It is very similar to what happens in a traditional individualized lesson with one tutor and one learner. In the learning oriented approach, technology needs to be adapted to the needs of learners and tutors to create suitable methods for working with it (Aberšek *et al.*, 2014). To this end, clustering of online learning resources is valuable to decision making in terms of tutorial strategies and instructional design.

6. Conclusions

A clustering algorithm for online learning resources is proposed based on the minimum spanning tree in this paper. Outliers are removed according to the density of resource which is measured by using Euclidean distance. A minimum spanning tree is generated to connect the neighboring online learning resources together by edges. The *K*-means clustering, average-link hierarchical clustering, complete-link hierarchical clustering algorithms and DBSCAN algorithm are tested with four data sets in order to evaluate the performance of different clustering techniques. Furthermore, it is elaborated how to apply the algorithm proposed in four different e-learning applications (i.e. classification of learner, discovery of learning path, recommendation of learning resource, and intelligent tutoring). The experimental results demonstrate the effectiveness of our algorithms proposed. Our technique will shed light on the real-world online learning, i.e., the minimum spanning tree based clustering algorithm can classify large amount of learning resources according to their characteristics. Such a kind of feature can reduce the time for searching of learning resources, alleviate the problem of ineffective studies, and improve the efficiency of online learners.

In the future, the density-based clustering method will be applied to choose the representative documents for sentiment analysis (Rao *et al.*, 2014). Moreover, the algorithm proposed will be further evaluated by using a large and high-dimensional learning corpus, as well as more real-world data sets. On the other hand, it will be valuable to conduct a longitudinal study.

References

Aberšek, B., Borstner, B. and Bregant, J. (2014), "The virtual science teacher as a hybrid system: cognitive science hand in hand with cybernetic pedagogy", *Journal of Baltic Science Education*, Vol. 13 No. 1, pp. 75-90.

Ben, S., Jin, Z. and Yang, J. (2011), "Guided fuzzy clustering with multi-prototypes", *Proceedings of the 2011 International Joint Conference on Neural Networks (IJCNN)*, *San Jose, CA*, *August*, pp. 2430-2436, doi: 10.1109/IJCNN.2011.6033534.

Boyer, S. and Veeramachaneni, K. (2015), "Transfer learning for predictive models in massive open online courses", *Proceedings of the 17th International Conference on Artificial Intelligence in Education (AIED)*, *Madrid*, *June*, pp. 54-63, doi: 10.1007/978-3-319-19773-9_6.

Deza, M. and Deza, E. (2016), *Encyclopedia of Distances*, ISBN 978-3-662-52844-0, 4th rev. ed., Springer-Verlag, Berlin Heidelberg.

Duan, L., Xiong, D., Lee, J. and Guo, F. (2006), "A local density based spatial clustering algorithm with noise", *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC'06)*, *Taipei*, *October*, pp. 4061-4066, doi: 10.1109/ICSMC.2006.384769.

Ester, M., Kriegel, H.P., Sander, J. and Xu, X. (1996), "A density-based algorithm for discovering clusters in large spatial databases with noise", *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, *Portland, OR*, *August*, pp. 226-231.

Ferschke, O., Yang, D., Tomar, G. and Rosé, C.P. (2015), "Positive impact of collaborative chat participation in an edX MOOC", *Proceedings of the 17th International Conference on Artificial Intelligence in Education (AIED)*, *Madrid*, *June*, pp. 115-124, doi: 10.1007/978-3-319-19773-9_12.

Govindarajan, K., Somasundaram, T.S., Kumar, V.S. and Kinshuk (2013), "Continuous clustering in big data learning analytics", *Proceedings of 2013 IEEE Fifth International Conference on Technology for Education (T4E)*, *December*, *Kharagpur*, pp. 61-64, doi: 10.1109/T4E.2013.23.

Jain, A.K. (2010), "Data clustering: 50 years beyond K-means", *Pattern Recognition Letters*, Vol. 31 No. 8, pp. 651-666.

Leung, E.W.C. and Li, Q. (2003), "A dynamic conceptual network mechanism for personalized study plan generation", *Proceedings of the 2nd International Conference on Web-based Learning (ICWL 2003)*, *Melbourne*, *August*, pp. 69-80, doi: 10.1007/978-3-540-45200-3_8.

Li, C.Z., Xu, Z.B. and Luo, T. (2013), "A heuristic hierarchical clustering based on multiple similarity measurements", *Pattern Recognition Letters*, Vol. 34 No. 2, pp. 155-162.

Li, Q., Lau, R.W.H., Wah, B., Ashman, H., Leung, E., Li, F. and Lee, V. (2009), "Guest editors' introduction: emerging internet technologies for e-learning", *IEEE Internet Computing*, Vol. 13 No. 4, pp. 11-17.

Luo, T., Zhong, C., Li, H. and Sun, X. (2010), "A multi-prototype clustering algorithm based on minimum spanning tree", *Proceedings of 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, *Yantai*, *August*, pp. 1602-1607, doi: 10.1109/FSKD.2010.5569359.

Manouselis, N., Vuorikari, R. and Assche, F.V. (2010), "Collaborative recommendation of e-learning resources: an experimental investigation", *Journal of Computer Assisted Learning*, Vol. 26 No. 4, pp. 227-242.

Mansur, A.B.F. and Yusof, N. (2013), "Social learning network analysis model to identify learning patterns using ontology clustering techniques and meaningful learning", *Computers & Education*, Vol. 63, April, pp. 73-86.

Mimaroglu, S. and Erdil, E. (2011), "Combining multiple clusterings using similarity graph", *Pattern Recognition*, Vol. 44 No. 3, pp. 694-703.

MITx and HarvardX (2014), "HarvardX-MITx Person-Course Academic Year 2013 De-Identified dataset, version 2.0", Harvard Dataverse, V10, doi: 10.7910/DVN/26147.

Päivinen, N. (2005), "Clustering with a minimum spanning tree of scale-free-like structure", *Pattern Recognition Letters*, Vol. 26 No. 7, pp. 921-930.

Phobun, P. and Vicheanpanya, J. (2010), "Adaptive intelligent tutoring systems for e-learning systems", *Procedia – Social and Behavioral Sciences*, Vol. 2 No. 2, pp. 4064-4069.

Prim, R.C. (1957), "Shortest connection networks and some generalizations", *Bell System Technical Journal*, Vol. 36 No. 6, pp. 1389-1401, doi: 10.1002/j.1538-7305.1957.tb01515.x.

Rao, Y.H. and Li, Q. (2012), "Term weighting schemes for emerging event detection", *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, *Macau*, *December*, pp. 105-112, doi: 10.1109/WI-IAT.2012.66.

Rao, Y.H., Li, Q., Mao, X.D. and Wenyin, L. (2014), "Sentiment topic models for social emotion mining", *Information Sciences*, Vol. 266, May, pp. 90-100.

Rossi, L.A. and Gnawali, O. (2014), "Language independent analysis and classification of discussion threads in coursera MOOC forums", *Proceedings of the IEEE International Conference on Information Reuse and Integration (IRI)*, *Redwood City, CA*, *August*, pp. 654-661, doi: 10.1109/IRI.2014.7051952.

Sabitha, A.S., Mehrotra, D. and Bansal, A. (2016), "Delivery of learning knowledge objects using fuzzy clustering", *Education and Information Technologies*, Vol. 21 No. 5, pp. 1329-1349.

Silva, P.F. da and Mustaro, P.N. (2009), "Clustering of learning objects with self-organizing maps", *Proceedings of IEEE Frontiers in Education Conference (FIE)*, *San Antonio, TX*, *October*, pp. 1-6, doi: 10.1109/FIE.2009.5350542.

Wang, S.Y., Tang, Z.P., Rao, Y.H., Xie, H.R. and Wang, F.L. (2015), "A clustering algorithm based on minimum spanning tree with e-learning applications", Current Developments in Web Based Learning: ICWL 2015 International Workshops, KMEL, IWUM, LA, Guangzhou, November, pp. 3-12, doi: 10.1007/978-3-319-32865-2_1.

Woldab, Z.E. (2014), "E-learning technology in pre-service teachers training-lessons for Ethiopia", *Journal of Educational and Social Research*, Vol. 4 No. 1, pp. 159-166, doi: 10.5901/jesr.2014.v4n1p159.

Xie, H.R., Li, Q. and Cai, Y. (2012), "Community-aware resource profiling for personalized search in folksonomy", *Journal of Computer Science and Technology*, Vol. 27 No. 3, pp. 599-610.

Xie, H.R., Li, Q., Mao, X.D., Li, X.D., Cai, Y. and Rao, Y.H. (2014), "Community-aware user profile enrichment in folksonomy", *Neural Networks*, Vol. 58, October, pp. 111-121.

Zbick, J. (2013), "A web-based approach for designing and deploying flexible learning tools", ICWE 2013 International Workshops ComposableWeb, QWE, MDWE, DMSSW, EMotions, CSE, SSN, and PhD Symposium, Aalborg, July, pp. 320-324, doi: 10.1007/978-3-319-04244-2_30.

Zou, D., Xie, H.R., Li, Q., Wang, F.L. and Chen, W. (2014), "The load-based learner profile for incidental word learning task generation", *Proceedings of the 13th International Conference on Web-based Learning (ICWL 2014)*, *Tallinn*, *August*, pp. 190-200, doi: 10.1007/978-3-319-09635-3_21.

Corresponding author
Fu Lee Wang can be contacted at: pwang@cihe.edu.hk