Effectiveness Analysis of The Application of Clustering in Student Grouping

Chen Xu[1, a], Li Zheng[2, b]

[1] Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China

[2] Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China

[a]email: xuchen.fx@gmail.com, [b]email: zhengli@tsinghua.edu.cn

Keywords: education data mining; clustering; student grouping

Abstract. In today's education practices, it is common to group students on their performance and teach them separately. Additionally, now a lot of data mining tools, including many education data mining tools, are available to teachers. With the help of these tools, student grouping can be more effective using clustering algorithms. This article proposes two principles of choosing clustering algorithm for student grouping. Two algorithms are chosen and an experiment is made to compare their effectiveness on students' learning data. Finally we give the conclusion on why one algorithm performs better than the other.

Introduction

Educational Data Mining (EDM) is the application of Data Mining techniques to educational data [1]. Recently, the increase in the use of e-learning systems has observed large repositories of data concerning how students learn, which is made available via the use of databases. The EDM process keeps teachers and educational researchers informed by turning the raw data into comprehensible information. This process include: preprocessing, data mining and postprocessing.

From the data mining tools and frameworks recently available, we find some commercial mining tools such as DBMiner and SPSS Clementine, as well as open source tools, e.g. Weka and Keel, are mainly designed for higher flexibility, not necessarily for specificity. Also, another feature we notice from a number of educational data mining tools, e.g. EPRules [2], E-learning Web Miner [3] and Pdinamet [4], is that most such tools are designed specifically for one or two functions, such as association or clustering. In these tools, association algorithms are commonly used to find interesting relationships of students' learning behavior and performance, and clustering algorithms are commonly used to group students with similar learning characteristics into several clusters.

This paper is organized as follows. The significance of grouping students is discussed in Section II. Next, in Section III we introduce common algorithms for clustering, select proper ones for teachers to use and choose a proper evaluation method to compare them. An experiment is made to show which of them has better performance and the results of it is in Section IV. Section V is our conclusions and lines of future work.

Significance of Grouping Students

In education practices, it is common to divide students into groups and teach them separately. It is generally believed that students may perform better if they are divided into groups according to their ability, for the learning materials can be geared to suit them better. This common practice is "stratified teaching". In this way, students in different levels can all have appropriate learning environment to make their own progress.

Although whether students in primary school and middle school should be grouped on their performance is controversial to some extent, it is certain that many courses in universities, such as English and computer, are being taken by many students with different levels of ability. There are even more such cases in adult education, spare-time schools and online courses. In these courses, grouping students are really necessary. In fact, many universities are giving courses of different

difficulties in such subjects for students.

However, it is not easy to divide students into different groups according to their ability. To meet this need, clustering algorithms can be used to measure the students' overall performance and facilitate the grouping of students. For this purpose, this paper aims at exploring the most appropriate algorithm and evaluation method to achieve the best effect.

Appropriate Algorithms and Evaluation Methods

Clustering methods can help us to group students, depending on the different learning activities. Clustering methods divide data objects into some subsets (called clusters), the objects in the same cluster are similar to each other. Using such algorithm, students in the same group together can do similar learning activities.

Clustering algorithms generally include four types, centroid-based, density-based, connectivity-based and distribution-based [7]. There are many different algorithms in each type. For our experiment, the following principles are adopted when we choose algorithms. First, the algorithm must be able to accept the number of clusters in the result as a parameter. In the application scenarios of grouping students, teachers usually decide the number of groups, thus a tool which cannot set the number of clusters is useless or confusing. Second, the parameters should be simple, intuitive, and comprehensible. No clustering algorithm can give perfect clusters on all data sets. Users can use the tool better, only when they better understand it. Considering that most teachers are not specialists in data mining, easier algorithms are more appropriate than complex ones. In the principles above, we don't choose the algorithm for processing large datasets. The dataset won't be too large in practice because generally the algorithm will only be used to group a class of no more than several hundreds of students.

Among these types, connectivity-based and distribution-based are more difficult to understand. Connectivity-based clustering usually includes a distance function and a linkage criterion, while distribution-based clustering is closely related to statistics and is based on distribution models. Very few EDM tools adopt these two algorithms, as they are hard to illustrate.

Density-based clustering defines clusters as areas with a much higher density than the rest of the data set. Algorithms study the density of points around each point to judge which points belong to the same cluster. The main advantage of this method is that it can find clusters of irregular shapes, e.g. S-shaped or ring-shaped clusters on a 2-D plane. Usually parameters characterizing when points around a point are dense, e.g. "there are t points within radius R from the point", are necessary. Popular algorithms are DBSCAN and OPTICS. Although the algorithms are easy to use, two limitations are unavoidable. First, the parameters are not intuitive, and then the number of clusters in the parameters can hardly be set.

Centroid-based clustering usually needs a number k meaning the number of clusters as a parameter, and divide n data objects into k clusters. Clusters are represented by their center, which may or may not be an object. An object belongs to the cluster represented by the nearest center - we can use Euclidian distance in this case. Typical centroid-based clustering algorithms are K-Means and K-Medoids. The two algorithms server our purposes well. A lot of tools currently choose K-Means for this function.

The following are the processes of the two algorithms. K-Means first:

INPUT: k, the number of clusters; D, a data set with n objects.

(1) Randomly choose k objects from D to be the centers of every cluster.

(2) Arrange each object to its nearest center to form k clusters.

(3) Calculate the average value, i.e. the shape center of every cluster, to be the new centers.

(4) If the k centers didn't change when executing (3), the algorithm end and output the k current clusters. Else go to (2).

Because it calculate the average values of objects in each cluster as new centers, the result may be inappropriate if there are outliers, i.e. the points which are numerically distant from the rest, in the data set.

K-Medoids is an approach to improve K-Means on this disadvantage, as follows:

INPUT: k, the number of clusters; D, a data set with n objects.

(1) Randomly choose k objects, o_1, …, o_k, from D to be the centers of every cluster.

(2) Arrange each object to its nearest center to form k clusters. Calculate the total cost of this division S=the sum of the distance of all objects to its cluster center.

(3) Check all objects except the centers, until an object o is found that if we use o to substitute a center oj, the new total cost S' will be less than S, or all objects have been checked. If we find such an object, use it to substitute oj and go to (2). If all objects have been checked, the algorithm end and output the k current clusters.

To compare these two algorithms, we need to select a way of evaluation. There exist supervised and unsupervised methods. The supervised methods are used to check the clustering result against a known classification, mainly to check its accuracy. The unsupervised methods are used in the circumstances when we don't know the classification of objects, to evaluate how separate the clusters are from each other and how tight each cluster is. In the application scenarios of grouping students, we cannot know which cluster is "most appropriate" for a student to be in, so the classification is unknown and only unsupervised methods are available. Although we hope to find an intuitive method to let non-specialists understand the quality of clustering results easily, there is no very intuitive method to use. Finally we selected a relatively easy and commonly used method, silhouette coefficient [6], to compare the two algorithms. It is defined as follows:

For a data set D with n objects, we suppose it is divided into k clusters C_1, … , C_k. For each object o, we calculate a(o), the average distance of o and the other objects in its cluster, and b(o), the minimal value of the average distances of o and the objects in the other clusters. That is, if $o \in C_i$, then

$$a(o) = \frac{\sum_{o' \in C_i, o \neq o'} \mathrm{dist}(o, o')}{|C_i| - 1} \qquad b(o) = \min_{C_j \neq C_i} \left\{ \frac{\sum_{o' \in C_j} \mathrm{dist}(o, o')}{|C_j|} \right\}$$

And the silhouette coefficient of object o is defined as

$$s(o) = \frac{b(o) - a(o)}{\max\{a(o), b(o)\}}$$

where a(o) means how well matched o is to the cluster it is assigned, the smaller the better, and b(o) means how separate o is from the other clusters, the larger the better. The quality of a clustering result can be represented by the average silhouette coefficient of all objects, whose value is between -1 to 1 and the larger the better.

Using Silhouette Coefficient to Compare Algorithms

Our experiments use silhouette coefficient to compare the performance of K-Means and K-Medoids algorithm on grouping students. Euclidian distance is used to measure the distance between objects, where numeric variables are all normalized that the maximum is cast to 1 and the minimum to 0. The non-numeric variables are considered to have a difference of 1 if unequal and 0 if equal. The data set of the experiments includes the sex, department (three categories: Art, Science or Engineering), average score of homework, homework submission, classroom attendance, score of class quiz. The total number of students is 171 and we consider the tow algorithms' performance when we divide them into 2-6 groups. The result of the experiment is shown in the following graph：

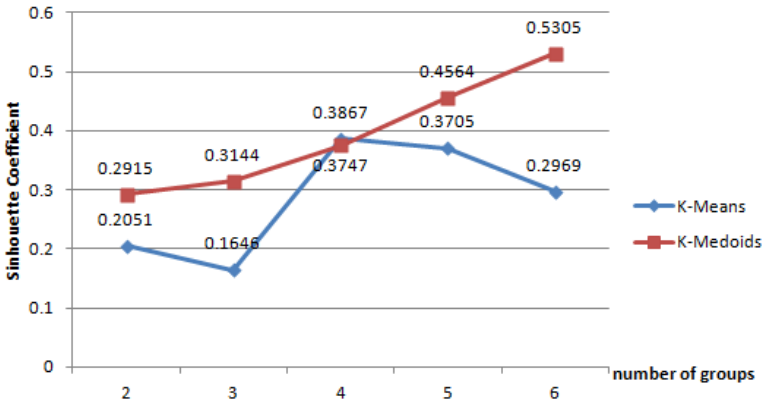

Figure 1. The Silhouette Coefficient of Clustering Results with Different Number of Groups.

From this analysis, we can see that the silhouette coefficients of K-Medoids algorithm tend to be higher than K-Means, and the silhouette coefficient increases with the increase of the number of groups, while the silhouette coefficient of K-Means algorithm often decreases. This means setting more groups sometimes mixes up the objects that should be assigned to different clusters. Considering the characteristic of educational data, such results may be caused by some outliers in the data of learners. For example, some learners may do much worse than the others. Using K-Medoids algorithm can reduce the impact of the outliers on overall grouping.

Conclusions

This paper first discussed and analyzed the demand of stratified teaching, and illustrated the necessities of using clustering in grouping students. For most users, easy to use and understand is a basic requirement. Thus we selected K-Means algorithm and K-Medoids algorithm and compared the two algorithms by the evaluation method of silhouette coefficient. The conclusion is that K-Medoids algorithm is more appropriate for grouping students.

However, the performance of clustering algorithm depends greatly on the specific characteristics of data sets and no algorithm fits all kinds of data sets. This article also proposes the following directions of further study: (1) including more data about the performance of students in online systems, such as access time to e-learning systems and (2) integrating the advantages of current algorithms and designing new ones to fit better the data of students' learning activities.

References

[1] C. Romero, S. Ventura. Educational data mining: A survey from 1995 to 2005[J]. Journal of Expert Systems with Applications, 2007(1-33): 135-146.

[2] C. Romero, S. Ventura, and P. De Bra. Knowledge discovery with genetic programming for providing feedback to courseware authors.[J]. User Modeling and User-Adapted Interaction, 2004(14-5): 425-464.

[3] M. Zorrilla and D. García-Saiz. A service oriented architecture to provide data mining services for non-expert data miners[J]. Decision Support Systems, in press

[4] E. Gaudioso, M. Montero, L. Talavera, and F. Hernandez-del-Olmo. Supporting teachers in collaborative student modeling: A framework and an implementation[J]. Expert System with Applications, 2009(36): 2260-2265.

[5] W. Xiang. Practical Research on Stratified Teaching of Basic Computer Courses in Universities[J]. Sichuan University of Arts and Science Journal, 2010(20-2): 119-121.

[6] J. Han, M. Kamber, and J. Pei. Data mining: concepts and techniques, 3rd ed[M]., Morgan Kaufmann, 2012, chapter. 10.

[7] Cluster analysis[EB/OL], <http://en.wikipedia.org/wiki/Cluster_analysis.