Knowledge-Based Systems 23 (2010) 144–149

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

G-ANMI: A mutual information based genetic clustering algorithm for categorical data

Shengchun Deng [a], Zengyou He [b,*], Xiaofei Xu [a]

[a] Department of Computer Science and Engineering, Harbin Institute of Technology, Harbin, China
[b] Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China

ARTICLE INFO

Article history:
Received 5 November 2008
Received in revised form 11 October 2009
Accepted 1 November 2009
Available online 10 November 2009

Keywords:
Clustering
Categorical data
Genetic algorithm
Mutual information
Cluster ensemble
Data mining

ABSTRACT

Identification of meaningful clusters from categorical data is one key problem in data mining. Recently, Average Normalized Mutual Information (ANMI) has been used to define categorical data clustering as an optimization problem. To find globally optimal or near-optimal partition determined by ANMI, a genetic clustering algorithm (G-ANMI) is proposed in this paper. Experimental results show that G-ANMI is superior or comparable to existing algorithms for clustering categorical data in terms of clustering accuracy.

1. Introduction

Cluster analysis aims at dividing data into meaningful clusters [1]. Clustering algorithms are increasingly required to deal with large-scale categorical data in real applications. To that end, a variety of clustering algorithms have been proposed for categorical data. A recent review on categorical data clustering algorithms can be found in [2].

In a categorical database, each attribute defines a partition in which data objects having same attribute value form a natural cluster. Based on this observation, the categorical data clustering problem has been explicitly formulated as a cluster ensemble problem in [3,4,2]. Information-theoretic methods were then used to define some optimization problems. In general, these optimization problems are NP-Complete. Therefore, research efforts towards this direction can be categorized by objective function and searching method.

On the basis of generalized conditional entropy, families of objective functions were generated [3]. To search possible partitions more efficiently, a genetic clustering algorithm (called ALG-RAND) was then presented.

Average Normalized Mutual Information (ANMI) was taken as the objective function in [4,2], which was initially proposed in the context of cluster ensemble [5]. To perform fast cluster analysis, graph partition algorithms and local searching algorithms were exploited in [4,2], respectively.

In the performance comparison conducted in [4,2], it has shown the advantage of ANMI against generalized conditional entropy in categorical data clustering. However, those algorithms in [4,2] tend to find local optimal partition and their capability in locating globally optimal or near-optimal partition is rather limited. To fulfill this void, this paper proposes a genetic clustering algorithm (called G-ANMI) for categorical data using ANMI as the objective function.

As shown in our experimental study, G-ANMI is able to obtain better clustering results than the algorithms in [4,2]. Meanwhile, G-ANMI has the following advantages against ALG-RAND [3]:

- G-ANMI can obtain better clustering results than ALG-RAND using less iterations and running times.
- In larger data sets, even when population size is relatively small, G-ANMI can find better solutions at the cost of more iterations.

The remainder of this paper is organized as follows. Section 2 introduces basic concepts and formulates the problem. In Section 3, we present the G-ANMI algorithm. Section 4 gives experimental results and Section 5 concludes the paper.

* Corresponding author.
E-mail addresses: zengyouhe@yahoo.com, eezyhe@ust.hk (Z. He).

0950-7051/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2009.11.001

2. Problem formulation

2.1. A unified cluster ensemble framework

A partition of n objects into k clusters can be represented as a set of k sets of objects $\{C_l|l=1,2,\ldots,k\}$ or as a label vector $\lambda \in N^n$. A *clusterer* Φ is a function that delivers a label vector given a set of objects, which provides a specific view of the data using a particular clustering algorithm.

Cluster ensemble (CE) is the method to combine several runs of different clustering algorithms to get a common partition of the original data set, aiming for consolidation of results from a portfolio of individual clustering results. More precisely, the basic idea of CE is to combine a set of r partitions $\lambda^{(1,2,\ldots,r)}$ into a single partition λ (the *consensus* partition) using a consensus function Γ [5].

Each categorical attribute defines a partition in which data objects sharing the same attribute value form a natural cluster. Hence, categorical data clustering problem can be regarded as a cluster ensemble problem. Formally, we have a categorical data set $D=\{X_1,X_2,\ldots,X_n\}$ with r attributes $A_1,A_2,\ldots,A_r$. Let V_i be the set of attribute values of A_i that are present in D, and $\Phi^{(i)}$ be the *clusterer* function that maps values in V_i to cluster labels. The optimal partition $\lambda^{(i)}$ determined by attribute A_i is defined as: $\lambda^{(i)}=(\Phi^{(i)}(X_j\cdot A_i)|X_j\cdot A_i\in V_i, X_j\in D\})$, where $1\leqslant i\leqslant r$, $1\leqslant j\leqslant n$ and $X_j\cdot A_i$ denotes the attribute value of X_j at A_i. Then, the problem of clustering categorical data is solved by combining these r partitions $\lambda^{(1,2,\ldots,r)}$ into a single partition λ using a specific consensus function Γ.

2.2. Objective function

A good combined partition should share as much information as possible with the given set of r partitions: $\Lambda=\{\lambda^{(q)}|q\in\{1,2,\ldots,r\}\}$. The consensus function Γ maps Λ to an integrated partition:

$$\Gamma:\{\lambda^{(q)}|q\in\{1,2,\ldots,r\}\}\rightarrow\lambda. \tag{1}$$

To determine how well the final partition summarizes the attribute partitions, Average Normalized Mutual Information (ANMI) [5] is exploited in this paper:

$$\phi^{(\mathrm{ANMI})}(\Lambda,\tilde{\lambda})=\frac{1}{r}\sum_{q=1}^{r}\phi^{(\mathrm{NMI})}(\lambda^{(q)},\tilde{\lambda}), \tag{2}$$

where $\phi^{(\mathrm{NMI})}(\lambda^{(q)},\tilde{\lambda})$ denotes the normalized mutual information between $\lambda^{(q)}$ and $\tilde{\lambda}$. Without loss of generality, normalized mutual information between two partitions $\lambda^{(a)}$ and $\lambda^{(b)}$ is computed as follows [5]:

$$\phi^{(\mathrm{NMI})}(\lambda^{(a)},\lambda^{(b)})=\frac{2}{n}\sum_{h=1}^{k^{(a)}}\sum_{g=1}^{k^{(b)}}n_g^{(h)}\log_{k^{(a)}k^{(b)}}\left(\frac{n_g^{(h)}n}{n^{(h)}n_g}\right), \tag{3}$$

where $k^{(a)}$ and $k^{(b)}$ are the number of clusters in partition $\lambda^{(a)}$ and $\lambda^{(b)}$, respectively. $n^{(h)}$ denotes the size of cluster C_h in partition $\lambda^{(a)}$, n_g denotes the size of cluster C_g in partition $\lambda^{(b)}$, and $n_g^{(h)}$ denotes the number of shared objects between C_h and C_g.

When the desired number of consensus clusters is k, the optimal combined partition $\lambda^{(k\text{-opt})}$ should have maximal ANMI:

$$\lambda^{(k\text{-opt})}=\arg\max_{\tilde{\lambda}}\phi^{(\mathrm{ANMI})}(\Lambda,\tilde{\lambda}), \tag{4}$$

where $\tilde{\lambda}$ goes through all possible k-partitions.

3. The G-ANMI algorithm

Genetic algorithm (GA) has shown good performance in numerous applications since its introduction by Holland [6].

In GA, solutions in the feasible search space are encoded in the forms of strings called chromosomes. A basic GA maintains a population of P chromosomes for some fixed population size P and evolves over generations. During each generation, three genetic operators, i.e. natural selection, crossover and mutation, are applied to the current population to produce a new population. Each chromosome in the population has a fitness value determined by the value of the objective function. Based on the principle of survival of the fittest, a few chromosomes in the current population are selected and each is assigned a number of copies, and then a new generation of chromosomes are yielded by applying crossover and mutation to the selected chromosomes.

There is a vast literature on GA, including studies on its theoretical and practical performance and many extensions of the basic algorithm. Although many sophisticated GA formulations exist, basic GA is employed in G-ANMI. The use of basic GA is based on the following considerations.

- Basic GA is simple and easy to implement, which requires less running time than those sophisticated algorithms in most cases.
- Basic GA is also adopted in [3]. The use of same genetic evolution procedure[1] provides solid basis for a fair performance comparison between G-ANMI and ALG-RAND.

The G-ANMI algorithm starts with a population of randomly selected partitions of objects, which are encoded as chromosomes. The fitness of each chromosome is evaluated using ANMI according to Eq. (2). Genetic evolution repeatedly changes the chromosomes in the current population to generate a new population. It is expected that chromosomes could be increasingly closer to the optimal partition with largest ANMI. Genetic procedure will halt when the best fitness in the current population is greater than the user-specified fitness threshold or there has been no relative improvement on best fitness after some consecutive iterations.

Since G-ANMI uses the same genetic procedure of ALG-RAND, the reader is referred to [3] for details on the working pipeline and required parameters.

4. Experimental results

Some categorical data sets obtained from the UCI Machine Learning Repository [7] were used to test the performance of different clustering algorithms. All algorithms were implemented in Java and all experiments were conducted on a Pentium4-2.4G machine with 512 M of RAM and running Windows 2000 Professional.

4.1. Real life data sets and evaluation method

Four data sets from UCI repository are used: voting, breast cancer,[2] zoo and mushroom. All these data sets contain only categorical attributes and class attribute. The information about the data sets is tabulated in Table 1. Note that the class attribute of the data has not been used in the clustering process.

[1] In our implementation, G-ANMI uses exactly the same genetic evolution procedure of ALG-RAND. The source codes of ALG-RAND are publicly available at: http://www.cs.umb.edu/~dana/GAClust/index.html.

[2] We use a data set that is slightly different from its original format in UCI, which has 683 objects. It is available at: http://research.cmis.csiro.au/rohanb/outliers/breast-cancer/brcancerall.dat.