A New Context-Based Clustering Framework for Categorical Data

Thanh-Phu Nguyen(✉), Duy-Tai Dinh, and Van-Nam Huynh

School of Knowledge Science, Japan Advanced Institute of Science and Technology,
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan
{ntphu,taidinh,huynh}@jaist.ac.jp

Abstract. Clustering is a fundamental task that has been utilized in many scientific fields, especially in machine learning and data mining. In clustering, dissimilarity measures play a key role in formulating clusters. For handling categorical values, the simple matching method is usually used for quantifying their dissimilarity. However, this method cannot capture the hidden semantic information that can be inferred from relationships among categories. In this paper, we propose a new clustering framework for categorical data that is capable of integrating not only the distributions of categories but also their mutual relationship information into the pattern proximity evaluation process of the clustering task. The effectiveness of the proposed clustering algorithm is proven by a comparative study conducted on existing clustering methods for categorical data.

Keywords: Clustering task · Categorical data
Dissimilarity measure · Unsupervised learning

1 Introduction

Clustering is a common method that is widely used in a variety of fields. Clustering groups data into clusters. For each cluster, objects in the same cluster are similar between themselves and dissimilar to objects in other clusters [1]. Clustering techniques can be classified into two main classes: hierarchical clustering and partitional clustering [2]. In fact, partitional methods have shown their effectiveness for solving clustering problems with scalability. Among them, the k-means [3] is probably the most well-known and widely used method. However, one inherent limitation of this approach is its data type constraint, as the k-means technically can only work with numerical data type.

During the last decade or so, several attempts have been made in order to remove the numeric data only limitation of k-means to make it applicable to clustering for categorical data. Particularly, some k-means like methods for categorical data have been proposed such as k-modes [4], k-representatives [5], k-centers [6] and k-means like clustering algorithm [7]. Although these algorithms

X. Geng and B.-H. Kang (Eds.): PRICAI 2018, LNAI 11012, pp. 697–709, 2018.
https://doi.org/10.1007/978-3-319-97304-3_53

use a similar clustering fashion to the k-means algorithm, they are different in defining cluster mean or dissimilarity measure for categorical data.

Furthermore, measures to quantify the dissimilarity (similarity) for categorical values are still not well-understood because there is no coherent metric available between categorical values thus far. Several methods have been proposed for encoding categorical data as numerical values such as dummy coding (or indicator coding) [8,9]. Particularly, they use binary values to indicate whether a categorical value is absent or present in a data record. However, by treating each category as an independent variable in that way, many important features and characteristics of categorical data type such as the distribution of categories or their relationships may not be taken into account.

Moreover, the significance of considering those information in order to quantify the dissimilarity between categorical data has been proven by several previous studies [7,10]. Especially, in the field of clustering with categorical data, most previous works have unfortunately neglected the semantic information potentially inferred from relationships among categories. In this research, we propose a new clustering algorithm that is able to integrate those kinds of information into the clustering process for categorical data.

Generally, the key contribution of this work is threefold:

- First, we extend an existing measure for categorical data to a new dissimilarity measure that is suitable for solving our problem. The measure is based on the conditional probability of correlated attributes values to include the mutual relationship between categorical attributes.
- Then, we propose a new categorical clustering algorithm that takes account of the semantic relationships between categories into the dissimilarity measure.
- Finally, we carry out an extensive experimental evaluation on benchmark data sets from UCI Machine Learning Repository [11] to evaluate the performance of our proposed algorithm with other existing methods in term of clustering quality.

The rest of this paper is organized as follows. In the Sect. 2, related work is reviewed. In the Sect. 3, a new clustering algorithm for categorical data is proposed. Next, the Sect. 4 describes an experimental evaluation. Finally, the Sect. 5 draws a conclusion.

2 Related Work

The conventional k-means algorithm [3] is one of the most often used clustering algorithms. It has some important properties: its computational complexity is linearly proportional to the size of datasets, it often terminates at a local optimum, its performance is dependent on the initialization of the centers [12]. However, working only on numerical data restricts some applications of the k-means algorithm. Specifically, it cannot process directly categorical data - a popular data type in many real life applications nowadays.

To address this limitation, several k-means like methods have been proposed for clustering task with categorical data. In 1997, Huang proposed k-modes and k-prototypes algorithms [4,13,14]. The k-modes [4] uses the simple matching measure to quantify the dissimilarity between categorical objects. It uses the modes to represent clusters, and a frequency-based method to update modes in the clustering process. The mode of a cluster is a data point whose attribute values are assigned by the most frequent values of the attribute's domain set appearing in the cluster. The k-modes first selects k initial modes (one for each cluster) from a data set X. Then, it allocates each object in X to a cluster whose mode is the nearest to that object based on the simple matching dissimilarity measure, and updates the mode of the cluster after each allocation. Next, it retests the dissimilarity of objects against the current modes after all objects have been allocated to clusters. Finally, it repeats the assignment step until no object has changed clusters after a full cycle test of the whole data set X.

The k-prototypes [13] is a combination of k-means and k-modes approaches to cluster objects with mixed numeric and categorical attributes. The k-prototypes can be divided into three steps: initial prototype selection, initial allocation and re-allocation. The first process randomly selects k objects as the initial prototypes for clusters. The second process assigns each object to a cluster and updates the cluster prototype after each assignment. The reallocation process updates prototypes for both the previous and current clusters of the object. Consequently, it iterates reallocation process until all objects are assigned to clusters and no object has changed clusters.

In 2004, San et al. proposed a k-means like algorithm named k-representatives [5]. The k-representatives applies the Cartesian product and union operations for the formation of cluster centers based on the notion of means in the numerical setting. In addition, it uses the dissimilarity based on the relative frequencies of categorical values within the cluster and the simple matching measure between categorical values. The algorithmic structure of k-representatives is formed in the same way as the k-modes [4].

Recently, Chen and Wang proposed a kernel-density-based clustering algorithm named k-centers [6]. The k-centers uses kernel density estimation method to define the mean of a categorical data set as a statistical center of the set. It incorporates a new feature weighting in which each attribute is automatically assigned with a weight to measure its individual contribution for the clusters. More recently, Nguyen and Huynh proposed the k-means like clustering framework [7]. This method extends k-representatives by replacing the simple matching measure with an information theoretic-based dissimilarity measure and adding a new concept of cluster centers.

3 The Proposed Clustering Algorithm

In this section, we propose a new clustering algorithm, namely $RICS$, that can integrate the mutual relationship information between categorical attributes into the clustering process. Details of the proposed algorithm are presented in two parts. The first part refers to significant elements of a k-means like clustering algorithm. The second part introduces a new dissimilarity measure for categorical data. Before going into the details of the clustering algorithm, we first introduce some notations that will be used in the rest of this paper.

3.1 Notations

Given a categorical data set X that contains n instances described by d attributes. The notations used in the rest of this paper are presented in the following.

- An attribute of X is denoted by A_j, $j \in \{1, \ldots, d\}$. For each A_j, its domain is denoted by $dom(A_j)$. Moreover, each value of A_j is denoted as a_l (or simply a) with $l \in \{1, \ldots, |dom(A_j)|\}$.
- An instance of X is presented as a vector $x = [x_1, \ldots, x_d]$ where the value of x at an attribute A_j is denoted as x_j, $j \in \{1, \ldots, d\}$.
- The frequency of $a_l \in dom(A_j)$ is denoted as $P(a_l)$ and calculated by

$$P(a_l) = \frac{count(A_j = a_l|X)}{|X|} \tag{1}$$

similarly, for $a_l \in dom(A_j)$ and $a_{l'} \in dom(A_{j'})$ we have

$$P(a_l, a_{l'}) = \frac{count((A_j = a_l) \text{ and } (A_{j'} = a_{l'})|X)}{|X|} \tag{2}$$

3.2 *k*-Means Like Clustering Framework

The clustering method proposed in this paper basically follows the k-means like clustering scheme as studied in [7]. Specifically, it still reserves the general procedure of the k-means but includes a modified concept of cluster centers based on the work of Chen and Wang [6] and a weighting method for each categorical attribute as well.

3.2.1 Representation of Cluster Centers

Let $C = \{C_1, \ldots, C_k\}$ be the set of k clusters of X, for any two different clusters C_i and $C_{i'}$ we have

$$C_i \cap C_{i'} = \emptyset \text{ if } i \neq i' \text{and } X = \bigcup_{i=1}^{k} C_i \tag{3}$$

Furthermore, for each cluster C_i, the center of C_i is defined as

$$V_i = [v_{i1}, \ldots, v_{ij}, \ldots, v_{id}] \tag{4}$$

where v_{ij} is a probability distribution on the domain of an attribute A_j that is estimated by a kernel density estimation function K.

$$v_{ij} = [p(a_1), \ldots, p(a_{|dom(A_j)|})] \tag{5}$$

where

$$p(a_l) = \sum_{a \in dom(A_j)} f_i(a) K(a|\lambda_j) \tag{6}$$

with $f_i(a)$ is the frequency probability of an attribute value a in the cluster V_i.

$$f_i(a) = \frac{count(A_j = a|V_i)}{|V_i|} \tag{7}$$

Moreover, consider σ_{ij} as the set that contains all available values of attribute A_j that exist in cluster V_i

$$\sigma_{ij} = \{a, a \in dom(A_j)|V_i\} \tag{8}$$

then the kernel function $K(a|\lambda_j)$ to estimate the probability of those attribute values in cluster V_i is defined as

$$K(a|\lambda_j) = \begin{cases} 1 - \frac{|\sigma_{ij}|-1}{|\sigma_{ij}|}\lambda_j & \text{if } a = a_l \\ \frac{1}{|\sigma_{ij}|}\lambda_j & \text{if } a \neq a_l \end{cases} \tag{9}$$

where λ_j is the smoothing parameter for C_j and has the value range of $[0, 1]$. In order to select the best parameter λ_j, the least squares cross validation (LSCV) method [6] is utilized. In the case $a \notin \sigma_{ij}$, $K(a|\lambda_j)$ value is set to 0.

Finally, from (4)–(6), we have the general formulation to compare the dissimilarity between a data instance $x \in X$ and a cluster center V_i described as below.

$$D(x, V_i) = \sum_{j=1}^{d} d(x_j, v_{ij}) = \sum_{j=1}^{d} \sum_{a \in dom(A_j)} p(a) \times dis(x_j, a) \tag{10}$$

where $dis(x_j, a)$ is the measure to quantify the dissimilarity between two values of an attribute A_j. Detailed information about this measure will be described in Subsect. 3.3.

3.2.2 Weighting Scheme for Categorical Attributes

A weighting model is also applied for categorical attributes as studied in [15]. Generally, a larger weight is set to attributes that have a smaller sum of within cluster distances and vice versa. More details of this method could be found in [15].

Consequently, we have a vector of weights $W = [w_1, \ldots, w_d]$ that is assigned to each attribute where each $w_j \leq 1$ and $\sum_{j=1}^{d} w_j = 1$.

Then, the weighted dissimilarity measure between a data instance x and a cluster center V_i could be defined as

$$D_w(x, V_i) = \sum_{j=1}^{d} w_j \times d(x_j, v_{ij}) \tag{11}$$

Based on these definitions, the clustering algorithm now aims to minimize the following objective function:

$$J(U,V,W) = \sum_{i=1}^{k}\sum_{g=1}^{n}\sum_{j=1}^{d} u_{i,g} \times w_j \times d(x_j, v_{ij}) \tag{12}$$

subject to

$$\begin{cases} \sum_{i=1}^{k} u_{i,g} = 1 & 1 \le g \le n \\ u_{i,g} \in \{0,1\} & 1 \le g \le n,\ 1 \le i \le k \\ \sum_{j=1}^{d} w_j = 1 & 0 \le w_j \le 1 \end{cases} \tag{13}$$

where $U = [u_{i,g}]_{n\times k}$ is the partition matrix. The algorithm for the k-means like clustering framework is described in Algorithm 1.

Algorithm 1. ***k*-means like clustering framework [7]**

Input: Data set $X = \{x_1, ..., x_n\}$
Output: Optimized clusters $C = \{C_1, ..., C_k\}$
1: Initialize centers for k clusters $V = [V_1, ..., V_k]$.
2: Initialize weights $W = [w_1, ..., w_d]$ and set $\lambda = 0$ for each attribute.
3: ***do***
4: Keep V and W fixed, generate U to minimize the distances between objects and cluster centers using Eq. (11).
5: Keep U fixed, update V using Eq. (5) and Eq. (6).
6: Generate W using formulas from [15].
7: ***while*** partitions still change.

3.3 A Context-Based Dissimilarity Measure for Categorical Data

In order to quantify the dissimilarity between categorical values, we extend the similarity measure proposed in [16] that could be able to integrate not only the distribution of categories but also their mutual relationship information. Specifically, the dissimilarity measure considers the amount of information to describe the appearances of pairs of attribute values rather than single values only. However, instead of considering all possible cases, only pairs of attributes that are highly correlated with each other are selected.

3.3.1 Correlation Analysis for Categorical Attributes

For the purpose of selecting highly correlated attribute pairs, the interdependence redundancy measure proposed by Au et al. [17] is adopted to quantify the dependency degree between each pair of attributes. Specifically, the interdependence redundancy value between two attributes A_j and $A_{j'}$ is computed as in the following formula.

$$R(A_j, A_{j'}) = \frac{I(A_j, A_{j'})}{H(A_j, A_{j'})} \tag{14}$$

where $I(A_j, A_{j'})$ denotes the mutual information [18] between attribute A_j and $A_{j'}$ and $H(A_j, A_{j'})$ is their joint entropy value. We have the formulas for those measures as the followings.

$$I(A_j, A_{j'}) = \sum_{p=1}^{|dom(A_j)|} \sum_{q=1}^{|dom(A_{j'})|} P(a_{jp}, a_{j'q}) * \log \frac{P(a_{jp}, a_{j'q})}{P(a_{jp}) * P(a_{j'q})} \tag{15}$$

$$H(A_j, A_{j'}) = - \sum_{p=1}^{|dom(A_j)|} \sum_{q=1}^{|dom(A_{j'})|} P(a_{jp}, a_{j'q}) * \log P(a_{jp}, a_{j'q}) \tag{16}$$

According to Au et al. [17], the interdependency redundancy measure has the value range of $[0, 1]$. A large value of R implies a high degree of dependency between attributes.

For each attribute A_j, in order to select its highly correlated attributes, a relation set is defined and denoted as S_j. Specifically, S_j contains attributes whose the associated interdependency redundancy values with A_j are larger than a specific threshold γ.

$$S_j = \{A_{j'} | R(A_j, A_{j'}) > \gamma, 1 \leq j, j' \leq d\} \tag{17}$$

3.3.2 New Dissimilarity Measure for Categorical Data

For integrating the relationship information that is contained in the set S_j, the conditional probability of correlated attributes values is utilized to include the mutual relationships between categorical attributes. In particular, to quantify the similarity between categorical values of attribute A_j, the following measure is implemented.

$$sim(x_j, x'_j) = \sum_{A_{j'} \in S_j} \sum_{a \in dom(A_{j'})} \frac{1}{|S_j|} \times \frac{1}{|dom(A_{j'})|} \times \frac{2 \times \log P(\{x_j, x'_j\}|a)}{\log P(x_j|a) + \log P(x'_j|a)} \tag{18}$$

It could be easily seen that the similarity measure in Eq. (18) have the value range of $[0, 1]$. Specifically, when x_j and x'_j are identical, their similarity degree is equal to 1. Then, the dissimilarity measure between two values of an attribute that is used in Eq. (10) could be defined as below.

$$dis(x_j, x'_j) = 1 - sim(x_j, x'_j) \tag{19}$$

The extended dissimilarity measure defined in Eq. (19) satisfies the following conditions:

1. $dis(x_j, x'_j) \geq 0$ for each x_j, x'_j with $j \in \{1, \ldots, d\}$
2. $dis(x_j, x_j) = 0$ with $\forall j \in \{1, \ldots, d\}$
3. $dis(x_j, x'_j) = dis(x'_j, x_j)$ for each x_j, x'_j with $j \in \{1, \ldots, d\}$.

For reducing the computational time of the proposed algorithm, the relation set of each attribute is generated in advance. Moreover, the dissimilarity between attribute values is also precomputed and cached in a multi-dimensional matrix for later used. Finally, details of the *RICS* algorithm are described in the Algorithm 2.

Algorithm 2. *RICS* clustering method

Input: Data set $X = \{x_1, ..., x_n\}$
Output: Optimized clusters $C = \{C_1, ..., C_k\}$
1: Generate relation set S_j for all attributes A_j using Eq. (14)-(17).
2: Precompute dissimilarity value $dis(a_l, a_{l'})$ for all $a_l, a_{l'} \in dom(A_j)$ with $j \in \{1, ..., d\}$ using Eq. (18), (19).
3: Initialize centers for k clusters $V = [V_1, ..., V_k]$.
4: Initialize weights $W = [w_1, ..., w_d]$ and set $\lambda = 0$ for each attribute.
5: ***do***
6: Keep V and W fixed, generate U to minimize the distances between objects and cluster centers using Eq. (11).
7: Keep U fixed, update V using Eq. (5) and Eq. (6).
8: Generate W using formulas from [15].
9: ***while*** partitions still change.

4 Experimental Evaluation

To evaluate the efficiency of the newly proposed algorithm, we conduct a comparative experiment on commonly used clustering methods for categorical data. Specifically, we contrast our new proposed clustering framework *RICS* with the implementation of k-modes [4], k-representatives [5] and k-means like clustering framework [7]. Furthermore, for each algorithm, we run 300 times per dataset. For the threshold value γ, it is practically found that with $\gamma = 0.1$ we could achieve general good results. Also, the value of parameter k is set equal to the number of classes in each dataset. The final results for three evaluation metrics are calculated by averaging the results of 300 running times.

4.1 Testing Datasets

Datasets for the experiment are selected from the UCI Machine Learning Repository [11]. The chosen 14 datasets contain not only categorical attributes but also integer and real values. For those numerical values, a discretization tool of Weka [19] is utilized for discretizing numerical values into equal intervals which are, in turn, treated as categorical values. In addition, the average dependency degree of each dataset is computed by averaging the interdependency redundancy values of all distinct pairs of attributes based on Eq. (14). Main characteristics of selected datasets are summarized in Table 1.

4.2 Clustering Quality Evaluation

In this research, in order to take advantages of class information in the original datasets, we take the same approach as [7] by utilizing the following supervised

Table 1. Main characteristics of 14 datasets from UCI

Dataset	Inst.	Attr.	Classes	Data types	Avg. dependency degree
Soybean	307	35	19	Categorical	0.153
Hayes-roth	160	5	3	Categorical	0.113
Wine	178	13	3	Integer, Real	0.089
Voting-records	435	16	2	Categorical	0.085
Dermatology	366	33	6	Categorical, integer	0.052
Breast-cancer	286	9	2	Categorical	0.027
Post-operative	90	8	3	Categorical, integer	0.014
Chess	3196	36	2	Categorical, integer	0.010
Tictactoe	958	9	2	Categorical	0.006
Splice	3190	61	3	Categorical	0.003
Car	1728	6	4	Categorical	0
Lenses	24	4	3	Categorical	0
Nursery	12960	8	5	Categorical	0
Balance-scale	625	4	3	Categorical	0

evaluation metrics for assessing clustering results: Purity, Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI).

In particular, let us denote $C = \{C_1, \ldots, C_J\}$ as a partition of the dataset that is generated by the clustering algorithm, and $P = \{P_1, \ldots, P_I\}$ is the partition which is inferred by the original class information. The total number of objects in the dataset is denoted by n.

Purity Metric. To compute purity value of a clustering result, firstly, each cluster is assigned to the class which is most frequent in the cluster. Then, the accuracy of this assignment is measured by counting the number of correctly assigned objects and dividing by the number of objects in the dataset. It is also worth noting that the purity metric could be significantly effected by the existence of imbalanced classes.

$$Purity(C, P) = \frac{1}{n} \sum_{j} \max_{i} |C_j \cap P_i| \tag{20}$$

NMI Metric. The NMI metric provides an information that is independent from the number of clusters [20]. This measure takes its maximum value when the clustering partition matches completely the original partition. NMI is computed as the average mutual information between any pair of clusters and classes.

$$NMI(C,P) = \frac{\sum_{i=1}^{I}\sum_{j=1}^{J}|C_j \cap P_i| \log \frac{n|C_j \cap P_i|}{|C_j||P_i|}}{\sqrt{\sum_{j=1}^{J}|C_j| \log \frac{|C_j|}{n} \sum_{i=1}^{I}|P_i| \log \frac{|P_i|}{n}}} \tag{21}$$

ARI Metric. The third metric is the Adjusted Rand Index [21]. Let a be the number of object pairs belonging to the same cluster in C and to the same class in P. This metric captures the deviation of a from its expected value corresponding to the hypothetical value of a obtained when C and P are two random, independent partitions.

The expected value of a is denoted as $E[a]$ and computed by:

$$E[a] = \frac{\pi(C)\pi(P)}{N(N-1)/2} \tag{22}$$

where $\pi(C)$ and $\pi(P)$ denote respectively the number of object pairs from the same clusters in C and from the same class in P. The maximum value for a is defined as:

$$max(a) = \frac{1}{2}(\pi(C) + \pi(P)) \tag{23}$$

The agreement between C and P can be estimated by the adjusted rand index as follows:

$$ARI(C,P) = \frac{a - E[a]}{max(a) - E[a]} \tag{24}$$

when $ARI(C,P) = 1$, we have identical partitions.

4.3 Experimental Results

From the results in Tables 2, 3 and 4, there is no best method for all of the testing datasets. However, we could see that the proposed clustering framework $RICS$ has achieved relatively good results comparing with other methods. Specifically, it performs effectively with highly correlated datasets such as soybean, hayes-roth, wine and dermatology. Moreover, the average results in all three tables show that our proposed framework has the best average results.

It is also worth noting that if we take a glance at the purity results in Table 2, the k-modes appears to outperform k-representatives and k-means like clustering method, and has a good performance when compared to $RICS$. However, when we make a more detailed inspection of the NMI and ARI results, it could be seen that k-modes actually has poor performances regarding those two more significant standards, while $RICS$ is still the one that has most of the best results over the total of 14 datasets.

Table 2. Purity results for categorical datasets

Data sets	RICS	k-means like framework	k-representatives	k-modes
Soybean	**0.7176**	0.7142	0.7152	0.6099
Hayes-roth	0.3954	0.3953	0.3998	**0.4079**
Wine	**0.9397**	0.9214	0.9380	0.7707
Voting-records	**0.8770**	0.8760	0.8764	0.8581
Dermatology	0.8560	0.8506	**0.8593**	0.7116
Breast-cancer	**0.7028**	**0.7028**	**0.7028**	**0.7028**
Post-operative	**0.7111**	**0.7111**	**0.7111**	**0.7111**
Chess	0.5223	0.5225	0.5222	**0.5761**
Tictactoe	0.6534	0.6534	0.6534	**0.6558**
Splice	**0.7586**	0.7572	0.6159	0.5188
Car	**0.7150**	0.7059	0.7046	0.7004
Lenses	0.6999	0.6981	**0.7018**	0.6446
Nursery	0.4449	0.4502	0.4324	**0.4704**
Balance-scale	0.5779	**0.5787**	0.5761	0.5496
Average	**0.6837**	0.6812	0.6721	0.6348

Table 3. NMI results for categorical datasets

Data sets	RICS	k-means like framework	k-representatives	k-modes
Soybean	0.7517	0.7473	**0.7545**	0.6069
Hayes-roth	0.0041	0.0038	0.0011	**0.0050**
Wine	0.7893	0.7580	**0.7941**	0.4252
Voting-records	**0.5055**	0.5002	0.4990	0.4359
Dermatology	**0.8551**	0.8512	**0.8551**	0.5735
Breast-cancer	**0.0041**	0.0040	0.0018	0.0038
Post-operative	0.0146	0.0140	0.0198	**0.0243**
Chess	0.0006	0.0007	0.0002	**0.0187**
Tictactoe	0.0346	**0.0393**	0.0087	0.0206
Splice	**0.4620**	0.4592	0.2820	0.0473
Car	**0.1435**	0.1234	0.1213	0.0475
Lenses	**0.3444**	0.3442	0.3432	0.1880
Nursery	0.0947	**0.1038**	0.0855	0.0601
Balance-scale	0.0485	**0.0491**	0.0474	0.0313
Average	**0.2895**	0.2856	0.2724	0.1777

Table 4. ARI results for categorical datasets

Data sets	RICS	k-means like framework	k-representatives	k-modes
Soybean	0.4642	0.4655	**0.4754**	0.3748
Hayes-roth	**−0.0102**	−0.0105	−0.0138	−0.0111
Wine	**0.8200**	0.7721	0.8145	0.4287
Voting-records	0.5642	0.5644	**0.5658**	0.5119
Dermatology	**0.7494**	0.7421	0.7389	0.5503
Breast-cancer	0.0018	0.0015	−0.0030	**0.0020**
Post-operative	**−0.0105**	−0.0113	−0.0110	−0.0178
Chess	−0.0001	0.0001	−0.0003	**0.0238**
Tictactoe	0.0325	**0.0380**	0.0218	0.0247
Splice	**0.3927**	0.3900	0.2021	0.0289
Car	0.0555	**0.0598**	0.0537	0.0239
Lenses	**0.2108**	0.2075	0.1835	0.0596
Nursery	0.0578	**0.0637**	0.0559	0.0506
Balance-scale	0.0507	**0.0522**	0.0505	0.0323
Average	**0.2413**	0.2382	0.2239	0.1487

5 Conclusion

In this paper, we have proposed a new clustering method for categorical data that could be able to integrate not only the distributions of categories but also their relationship information into the quantification of dissimilarity between data objects. The experiments have shown that the proposed clustering algorithm $RICS$ has a competitive performance when compared to other popular used clustering methods for categorical data. For the future work, we are planning to extend $RICS$ so that it could be used to solve the problem of clustering with mixed numeric and categorical datasets.

References

1. Berkhin, P.: A survey of clustering data mining techniques. In: Kogan, J., Nicholas, C., Teboulle, M. (eds.) Grouping Multidimensional Data, pp. 25–71. Springer, Heidelberg (2006). https://doi.org/10.1007/3-540-28349-8_2
2. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Comput. Surv. (CSUR) **31**(3), 264–323 (1999)
3. MacQueen, J.: Some methods for classification and analysis of multivariate observations. The Regents of the University of California (1967)
4. Huang, Z.: Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min. Knowl. Discov. **2**(3), 283–304 (1998)
5. San, O.M., Huynh, V.N., Nakamori, Y.: An alternative extension of the k-means algorithm for clustering categorical data. Int. J. Appl. Math. Comput. Sci. **14**, 241–247 (2004)

6. Chen, L., Wang, S.: Central clustering of categorical data with automated feature weighting. In: Twenty-Third International Joint Conference on Artificial Intelligence (2013)
7. Nguyen, T.-H.T., Huynh, V.-N.: A k-means-like algorithm for clustering categorical data using an information theoretic-based dissimilarity measure. In: Gyssens, M., Simari, G. (eds.) FoIKS 2016. LNCS, vol. 9616, pp. 115–130. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30024-5_7
8. Cohen, J., Cohen, P.: Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences. L. Erlbaum Associates, Hillsdale (1983)
9. Ralambondrainy, H.: A conceptual version of the K-means algorithm. Pattern Recogn. Lett. **16**(11), 1147–1157 (1995)
10. Ienco, D., Pensa, R.G., Meo, R.: From context to distance: learning dissimilarity for categorical data clustering. ACM Trans. Knowl. Discov. Data **6**(1), 1:1–1:25 (2012)
11. Lichman, M.: UCI machine learning repository (2013). http://archive.ics.uci.edu/ml
12. Gan, G., Ma, C., Wu, J.: Data Clustering: Theory, Algorithms, and Applications, vol. 20. SIAM, Philadelphia (2007)
13. Huang, Z.: Clustering large data sets with mixed numeric and categorical values. In: Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), pp. 21–34. Citeseer (1997)
14. Huang, Z.: A fast clustering algorithm to cluster very large categorical data sets in data mining. In. In Research Issues on Data Mining and Knowledge Discovery, pp. 1–8 (1997)
15. Huang, J.Z., Ng, M.K., Rong, H., Li, Z.: Automated variable weighting in k-means type clustering. IEEE Trans. Pattern Anal. Mach. Intell. **27**(5), 657–668 (2005)
16. Nguyen, T.-P., Ryoke, M., Huynh, V.-N.: A new context-based similarity measure for categorical data using information theory. In: Huynh, V.-N., Inuiguchi, M., Tran, D.H., Denoeux, T. (eds.) IUKM 2018. LNCS (LNAI), vol. 10758, pp. 114–125. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75429-1_10
17. Au, W.H., Chan, K.C.C., Wong, A.K.C., Wang, Y.: Attribute clustering for grouping, selection, and classification of gene expression data. IEEE/ACM Trans. Comput. Biol. Bioinf. **2**(2), 83–101 (2005)
18. MacKay, D.J.C.: Information Theory Inference and Learning Algorithms. Cambridge University Press, New York (2002)
19. Hall, M.A., Holmes, G.: Benchmarking attribute selection techniques for discrete class data mining. IEEE Trans. Knowl. Data Eng. **15**(6), 1437–1447 (2003)
20. Strehl, A., Ghosh, J.: Cluster ensembles a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. **3**, 583–617 (2003)
21. Hubert, L., Arabie, P.: Comparing partitions. J. Classif. **2**(1), 193–218 (1985)