ISSN 22773061

A Bootstrap Aggregating Technique on Link-Based Cluster Ensemble Approach for Categorical Data Clustering

S Pavan Kumar Reddy*, U Sesadri
M Tech (CSE) *
HOD of CSE
VITS - PDTR
pavansana8@gmail.com

Abstract:

Although attempts have been made to solve the problem of clustering categorical data via cluster ensembles, with the results being competitive to conventional algorithms, it is observed that these techniques unfortunately generate a final data partition based on incomplete information. The underlying ensemble-information matrix presents only cluster-data point relations, with many entries being left unknown. The paper presents an analysis that suggests this problem degrades the quality of the clustering result, and it presents a BSA (Bootstrap Aggregation) is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy along with a new link-based approach, which improves the conventional matrix by discovering unknown entries through similarity between clusters in an ensemble. In particular, an efficient BSA and link-based algorithm is proposed for the underlying similarity assessment. Afterward, to obtain the final clustering result, a graph partitioning technique is applied to a weighted bipartite graph that is formulated from the refined matrix. Experimental results on multiple real data sets suggest that the proposed link-based method almost always outperforms both conventional clustering algorithms for categorical data and well-known cluster ensemble techniques.

Index Terms—BSA, Clustering, categorical data, cluster ensembles, link-based similarity, data mining.

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol 10, No.8

editor@cirworld.com

www.cirworld.com, member.cirworld.com

1. INTRODUCTION

Bootstrap aggregating (bagging) is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. It also reduces variance and helps to avoid over fitting. Although it is usually applied to decision tree methods, it can be used with any type of method. Bagging is a special case of the model averaging approach.

Given a standard training set D of size n, bagging generates m new training sets D_i, each of size n′ < n, by sampling from D uniformly and with replacement. By sampling with replacement, some observations may be repeated in each D_i. If n′=n, then for large n the set D_i is expected to have the fraction (1 - 1/e) (≈63.2%) of the unique examples of D, the rest being duplicates.[1] This kind of sample is known as a bootstrap sample. The m models are fitted using the above m bootstrap samples and combined by averaging the output (for regression) or voting (for classification).

Bagging leads to "improvements for unstable procedures" (Breiman, 1996), which include, for example, neural nets, classification and regression trees, and subset selection in linear regression (Breiman, 1994). On the other hand, it can mildly degrade the performance of stable methods such as K-nearest neighbors (Breiman, 1996).

Data clustering is one of the fundamental tools we have for understanding the structure of a data set. It plays a crucial, foundational role in machine learning, data mining, information retrieval, and pattern recognition. Clustering aims to categorize data into groups or clusters such that the data in the same cluster are more similar to each other than to those in different clusters. Many well-established clustering algorithms, such as k-means [1] and PAM [2], have been designed for numerical data, whose inherent properties can be naturally employed to measure a distance (e.g., Euclidean) between feature vectors [3], [4]. However, these cannot be directly applied for clustering of categorical data, where domain values are discrete and have no ordering defined. An example of categorical attribute is sex ¼ f emale; female g or shape ¼ f circle; rectangle . . .g.

As a result, many categorical data clustering algorithms have been introduced in recent years, with applications to interesting domains such as protein interaction data [5]. The initial method was developed in [6] by making use of Gower's similarity coefficient [7].

1.1 Example of Ozone Data:

To illustrate the basic principles of bagging, below is an analysis on the relationship between ozone and temperature (data from Rous see u w and Leroy (1986), available at classic data sets, analysis done in R).

The relationship between temperature and ozone in this data set is apparently non-linear, based on the scatter plot. To mathematically describe this relationship, LOESS smoothers (with span 0.5) are used. Instead of building a single smoother from the complete data set, 100 bootstrap samples of the data were drawn. Each sample is different from the original data set, yet resembles it in distribution and variability. For each bootstrap sample, a LOESS smoother was fit. Predictions from these 100 smoothers were then made across the range of the data. The first 10 predicted smooth fits appear as grey lines in the figure below. The lines are clearly very wiggly and they over fit the data - a result of the span being too low.

But taking the average of 100 smoothers, each fitted to a subset of the original data set, we arrive at one bagged predictor (red line). Clearly, the mean is more stable and there is less over fit.

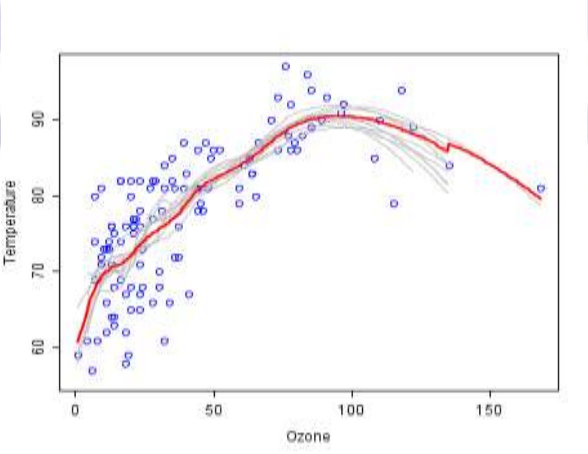


Figure 1: Ozone Data Analysis

Following that, the k-modes algorithm in [8] extended the conventional k-means with a simple matching dissimilarity measure and a frequency-based method to update centroids (i.e., clusters' representative). As a single-pass algorithm, Squeezer [9] makes use of a prespecified similarity threshold to determine which of the existing clusters (or a new cluster) to which a data point under examination is assigned. LIMBO [10] is a hierarchical clustering algorithm that uses the Information Bottleneck (IB) framework to define a distance measure for categorical tuples.

The concepts of evolutionary computing and genetic algorithm have also been adopted by a partitioning method for categorical data, i.e., G A Clust [11]. Cobweb [12] is a model-based method primarily exploited for categorical data sets. Different graph models have also been investigated by the STIRR [13], ROCK [14], and CLICK [15] techniques. In addition, several density-based algorithms have also been devised for such purpose, for instance, CACTUS [16], COOLCAT [17], and CLOPE [18].

Although, a large number of algorithms have been introduced for clustering categorical data, the No Free Lunch theorem [19] suggests1 there is no single clustering algorithm that performs best for all data sets [20] and can discover all types of cluster shapes and structures presented in data [21]. Each algorithm has its own strengths and weaknesses. For a particular data set, different algorithms, or even the same algorithm with different parameters, usually provide distinct solutions. Therefore, it is difficult for users to decide which algorithm would be the proper alternative for a given set of data. Recently, cluster ensembles have emerged as an effective solution that is able to overcome these limitations, and improve the robustness as well as the quality of clustering results. The main objective of cluster ensembles is to combine different clustering decisions in such a way as to achieve accuracy superior to that of any individual clustering. Examples of well-known ensemble methods are:

1. the feature-based approach that transforms the problem of cluster ensembles to clustering categorical data (i.e., cluster labels) [11], [22], [23], [24],

2. the direct approach that finds the final partition through relabeling the base clustering results [25], [26],

3. graph-based algorithms that employ a graph partitioning methodology [27], [28],

4. The pair wise-similarity approach that makes use of co-occurrence relations between data points.

Despite notable success, these methods generate the final data partition based on incomplete information of a cluster ensemble. The underlying ensemble- information matrix presents only cluster-data point relationships while completely ignores those among clusters. As a result, the performance of existing cluster ensemble techniques may consequently be degraded as many matrix entries are left Unknown. This paper introduces a link-based approach to refining the aforementioned matrix, giving substantially less unknown entries. A link-based similarity measure is exploited to estimate unknown values from a link network of clusters. This research uniquely bridges the gap between the task of data clustering and that of link analysis. It also enhances the capability of ensemble methodology for categorical data, which has not received much attention in the literature. In addition to the problem of clustering categorical data that is investigated herein, the proposed framework is generic such that it can also be effectively applied to other data types.

The rest of this paper is organized as follows: Section 2 presents the cluster ensemble framework upon which the current research has been established. The proposed link based approach, including the underlying intuition of refining an ensemble-information matrix and details of a link-based.

2. CLUSTER ENSEMBLE METHODOLOGIES

2.1 Problem Formulation and General Framework

Let X ¼ fx1; . . . ; x; Ng be a set of N data points and ¼ f1; . . . ; Mg be a cluster ensemble with M base clustering's, each of which is referred to as an ensemble member. Each base clustering returns a set of clusters i ¼ fCi 1; Ci j ¼ X, where ki is the number of clusters in the ith clustering. For each x 2 X, Cx denotes the cluster label to which the data point x belongs. In the I th clustering, Cx ¼ "j" (or "Ci j") if x 2 C,I,j. The problem is to find a new partition of a data set X that summarizes the information from the cluster ensemble _. Fig. 1 shows the general framework of cluster ensembles. Essentially, solutions achieved from different base clustering are aggregated to form a final partition. This Meta level methodology involves two major tasks of: 1) generating a cluster ensemble, and 2) producing the final partition, normally referred to as a consensus function.

2.2 Ensemble Generation Methods

It has been shown that ensembles are most effective when constructed from a set of predictors whose errors are dissimilar. To a great extent, diversity among ensemble members is introduced to enhance the result of an ensemble. Particularly for data clustering, the results obtained with any single algorithm over much iteration are usually very similar. In such a circumstance where all ensemble members agree on how a data set should be partitioned, aggregating the base clustering results will show no improvement over any of the constituent members. As a result, several heuristics have been proposed to introduce artificial instabilities in clustering algorithms, giving diversity within a cluster ensemble. The following ensemble generation methods yield different clustering's of the same data, by exploiting different cluster models and different data partitions.

Homogeneous ensembles. Base clustering's are created using repeated runs of a single clustering algorithm, with several sets of parameter initializations, such as cluster centers of the k-means clustering technique. Random-k. One of the most successful techniques is randomly selecting the number of clusters (k) for each ensemble member. Data subspace/sampling. Cluster ensemble can also be achieved by generating base clustering's from different subsets of initial data. It is intuitively assumed that each clustering algorithm will provide different levels of performance for different partitions of a data set [27]. Practically speaking, data partitions are obtained by projecting data onto different subspaces [24], choosing different subsets of features [29], or data sampling.

Heterogeneous ensembles. A number of different clustering algorithms are used together to generate base clustering's. Mixed heuristics. In addition to using one of the aforementioned methods, any combination of them can be applied as well

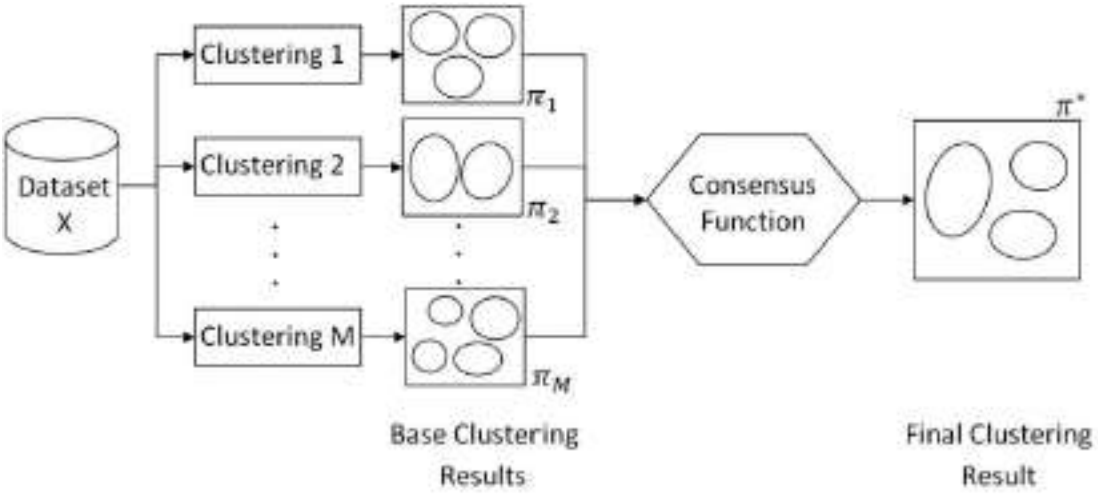


Figure2: The basic process of cluster ensembles

It first applies multiple base clustering's to a data set X to obtain diverse clustering decisions (1 . . . M). Then, these solutions are combined to establish the final Clustering result using a consensus function.

2.3 Cluster Ensembles of Categorical Data

While a large number of cluster ensemble techniques for numerical data have been put forward in the previous decade, there are only a few studies that apply such a methodology to categorical data clustering. The method introduced to creates an ensemble by applying a conventional clustering algorithm (e.g., k-modes [8] and COOLCAT [17]) to different data partitions, each of which is constituted by a unique subset of data attributes. Once an ensemble has been obtained, the graph-based consensus functions of [28] are utilized to generate the final clustering result.

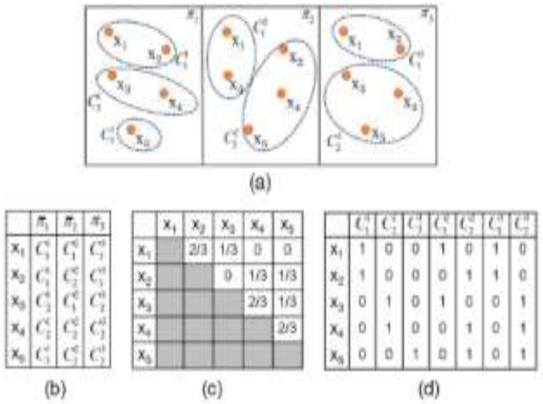
Unlike the conventional approach, the technique developed acquires a cluster ensemble without actually implementing any base clustering on the examined data set.

In fact, each attribute is considered as a base clustering that provides a unique data partition. In particular, a cluster in such attribute-specific partition contains data points that share a specific attribute value (i.e., categorical label). Thus, the ensemble size is determined by the number of categorical labels, across all data attributes. The final clustering result is generated using the graph-based consensus techniques presented in [29]. Specific to this so-called "direct" ensemble generation method, a given categorical data set can be represented using a binary cluster-association matrix, whose example is shown earlier in Fig. 3d. Such an information matrix is analogous to the "market-basket" numerical representation of categorical data, which has been the focus of traditional categorical data analysis.

3. A NOVEL LINK-BASED APPROACH

Existing cluster ensemble methods to categorical data analysis rely on the typical pair wise-similarity and binary cluster-association matrices which summarize the underlying ensemble information at a rather coarse level.

Many matrix entries are left "unknown" and simply recorded as "0." Regardless of a consensus function, the quality of the final clustering result may be degraded. As a result, a link based method has been established with the ability to discover unknown values and, hence, improve the accuracy of the ultimate data partition. In spite of promising findings, this initial framework is based on the data point data point pair wise-similarity matrix, which is highly expensive to obtain. The link-based similarity technique, Sim Rank that is employed to estimate the similarity among data points is inapplicable to a large data set.



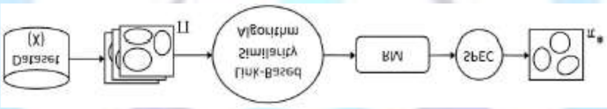
	π_1	π_2	π_3
x_1	C_1^1	C_1^2	C_1^3
x_2	C_1^1	C_2^2	C_1^3
x_3	C_2^1	C_1^2	C_2^3
x_4	C_2^1	C_2^2	C_2^3
x_5	C_3^1	C_2^2	C_2^3

	x_1	x_2	x_3	x_4	x_5
x_1		2/3	1/3	0	0
x_2			0	1/3	1/3
x_3				2/3	1/3
x_4					2/3
x_5					

	C_1^1	C_2^1	C_3^1	C_1^2	C_2^2	C_1^3	C_2^3
x_1	1	0	0	1	0	1	0
x_2	1	0	0	0	1	1	0
x_3	0	1	0	1	0	0	1
x_4	0	1	0	0	1	0	1
x_5	0	0	1	0	1	0	1

Figure3: (a) cluster ensemble and the corresponding (b) label-assignment matrix, (c) pair wise-similarity matrix, and (d) binary cluster-association matrix, respectively.

To overcome these problems, a new link-based cluster ensemble (LCE) approach is introduced herein. It is more efficient than the former model, where a BM-like matrix is Figure 3 Examples of (a) cluster ensemble and the corresponding (b) label-assignment matrix, (c) pair wise-similarity matrix, and (d) binary cluster-association matrix, respectively to represent the ensemble information. The focus has shifted from revealing the similarity among data points to estimating those between clusters. A new link-based algorithm has been specifically proposed to generate such measures in an accurate, inexpensive manner.



The LCE methodology is illustrated in Fig. 4. It includes three major steps of: 1) creating base clustering's to form a cluster ensemble generating a refined cluster-association matrix (RM) using a link-based similarity algorithm, and Producing the final data partition by exploiting the spectral graph partitioning technique as a consensus function.

3.1 Creating a Cluster Ensemble

Type I (Direct ensemble). Following, the first type of cluster ensemble transforms the problem of categorical data clustering to cluster ensembles by considering each categorical attribute value (or label) as a cluster in an ensemble. Let X ¼ fx1; . . . ; xNg be a set of N data points, A ¼ f1; . . . ; be a set of categorical attributes, and ¼ f 1; . . . ; Mg be a set of M partitions. Each partition i is generated for a specific categorical attribute a i 2 A. Clusters belonging to a partition i ¼ f; Ci;1; correspond to different values of the attribute values of attribute a i. With this formalism, categorical data X can be directly transformed to a cluster ensemble without actually implementing any base clustering. While single-attribute data partitions may not be as accurate as those obtained from the clustering of all data attributes, they can bring about great diversity within an ensemble. Besides its efficiency, this ensemble generation method has the potential to lead to a high-quality clustering result.

3.2 Generating a Refined Matrix

Several cluster ensemble methods, both for numerical [28], and categorical data are based on the binary cluster-association matrix. Each entry in this matrix 1represents a crisp association degree between data point xi 2 X and cluster 2. According to Fig. 3 that shows an example of cluster ensemble and the corresponding BM, a large number of entries in the BM are unknown, each presented with "0." Such condition occurs when relations between different clusters of a base clustering are originally assumed to be nil. In fact, each data point can possibly associate (to a certain degree within ½ to several clusters of any particular clustering. These hidden or unknown associations can be estimated from the similarity among clusters, discovered from a network of clusters.

Based on this insight, the refined cluster-association matrix is put forward as the enhanced variation of the original BM. Its aim is to approximate the value of unknown associations ("0") from known ones ("1"), whose association degrees are preserved within the RM, i.e. For each clustering t; t ¼ 1 . . .M and their corresponding clusters Ct 1; . . . ; Ct kt (where kt is the number of clusters in the clustering t), the association degree RM xi; cl 2 ½0; 1 that data point xi 2 X has with each cluster cl 2 is a cluster label (corresponding to a particular cluster of the clustering to which data point xi belongs. In addition, sim Cx; Cy 2 ½0; 1 denotes the similarity between any two clusters C x; Cy, which can be discovered using the following link-based algorithm. Note that, for any clustering .Unlike the measure of fuzzy membership, the typical constraint of ¼ 1 is not appropriate for rescaling associations within the RM. In fact, this local normalization will significantly distort the true semantics of known associations ("1"), such that their magnitudes become dissimilar, different from one clustering to another. According to the empirical investigation, this fuzzy-like enforcement decreases the quality of the RM, and hence, the performance of the resulting cluster ensemble method.

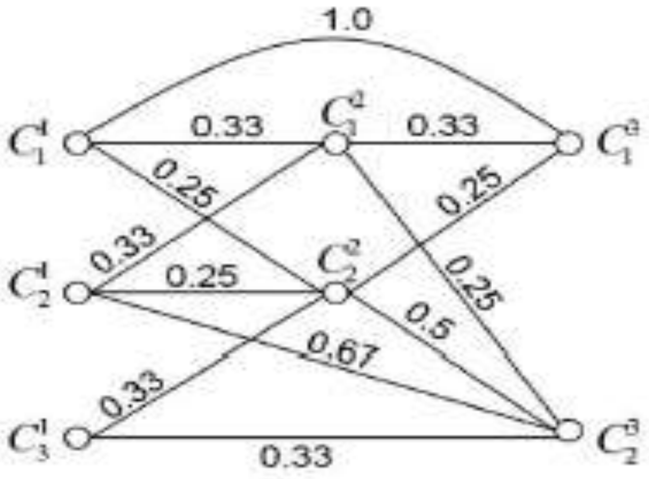


Figure5: The link-based cluster ensemble framework

3.2.1 Weighted Triple-Quality (WTQ): A New Link-Based Similarity Algorithm

Given a cluster ensemble _ of a set of data points X, a weighted graph G ¼ V ;W can be constructed, where V is the set of vertices each representing a cluster and W is a set of weighted edges between clusters. Formally, the weight assigned to the edge w,x, y to W, that connects clusters Cx; Cy 2 V , is estimated by the proportion of their overlapping where X denotes the set of data points belonging to cluster Cz V . Fig. 4 shows the network of clusters that is generated from the example given in Fig. 2. Note that circle nodes represent clusters and edges exist only when the corresponding weights are nonzero.

Shared neighbors have been widely recognized as the basic evidence to justify the similarity among vertices in a link network. Formally, a vertex Ck 2 V is a common neighbor (sometimes called "triple," which is short for "center of the connected triple") of vertices Cx; Cy 2 V, provided that wxk; wyk 2 W. Many advanced methods extend this basis by taking into account common neighbors that may be many edges away from the two under examination: for instance, Connected-Path,

4. PERFORMANCE EVALUATIONS

4.1 Investigated Data Sets

The experimental evaluation is conducted over nine data sets. The "20Newsgroup" data set is a subset of the well known text data collection 20-Newsgroups,2 while the others are obtained from the UCI Machine Learning Repository . Their details are summarized in Table 1.

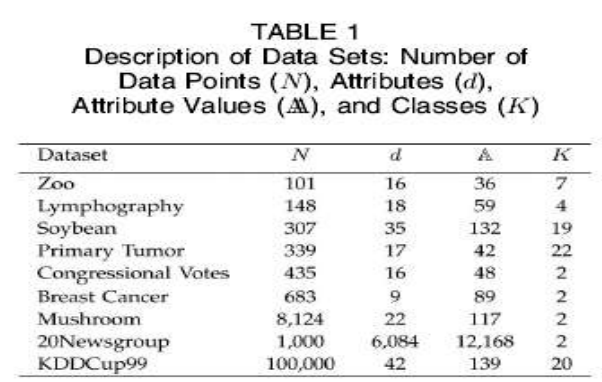


TABLE 1
Description of Data Sets: Number of Data Points (N), Attributes (d), Attribute Values ($\mathbb{A}$), and Classes (K)

Dataset	N	d	$\mathbb{A}$	K
Zoo	101	16	36	7
Lymphography	148	18	59	4
Soybean	307	35	132	19
Primary Tumor	339	17	42	22
Congressional Votes	435	16	48	2
Breast Cancer	683	9	89	2
Mushroom	8,124	22	117	2
20Newsgroup	1,000	6,084	12,168	2
KDDCup99	100,000	42	139	20

Missing values (denoted as "?") in these data sets are simply treated as a new categorical value. The "20Newsgroup" data set contains 1,000 documents from two newsgroups, each of which is described by the occurrences of 6,084 different terms. In particular, the frequency (f 2 f0; 1; . . . ;1 g) that a key word appears in each document is transformed into a nominal value: "Yes" if f > 0, "No" otherwise. Moreover, the "KDDCup99" data set used in this evaluation is a randomly selected subset of the original data. Each data point (or record) corresponds to a network connection and contains 42 attributes: some are nominal and the rest are continuous. Following the study in [17], numerical attributes are transformed to categorical using a simple discretization process. For each attribute, any value less than the median is assigned a label "0," otherwise "1." Note that the selected set of data records covers 20 different connection classes. These two data sets are specifically included to assess the performance of different clustering methods, with respect to the large numbers of dimensionality and data points, respectively.

4.2 Experiment Design

The experiments set out to investigate the performance of LCE compared to a number of clustering algorithms, both specifically developed for categorical data analysis and those state-of-the-art cluster ensemble techniques found in literature. Baseline model is also included in the assessment, which simply applies SPEC, as a consensus function, to the conventional BM (see Section 4.2.2). For comparison, as in [28] each clustering method divides data points into a partition of K (the number of true classes for each data set) clusters, which is then evaluated against the corresponding true partition using the following set of label-based evaluation indices: Classification Accuracy (CA) [23], Normalized Mutual Information (NMI) and Adjusted Rand (AR) Index . Further details of these quality measures are provided in Section I of the online supplementary.3 Note that, true classes are known for all data sets but are explicitly not used by the cluster ensemble process. They are only used to evaluate the quality of the clustering results.

4.2.1 Parameter Settings

In order to evaluate the quality of cluster ensemble methods previously identified, they are empirically compared, using the settings of cluster ensembles exhibited below. Five types of cluster ensembles are investigated in this evaluation: Type-I, Type-II (Fixed-k), Type-II (Random-k), Type-III (Fixed-k), and Type-III (Random- k). The k-modes clustering algorithm is specifically used to generate the base clustering's. Ensemble size (M) of 10 is experimented. The quality of each method with respect to a specific ensemble setting is generalized as the average of 50 runs. The constant decay factor (DC) of 0.9 is exploited with WTQ. Performance among assessed ensemble methods. In addition, Cobweb is the most effective among five categorical data clustering algorithms included in this evaluation.

Similar experimental results are also observed using NMI and AR evaluation indices. The corresponding details are given in Section II-A of the online supplementary. In order to further evaluate the quality of identified techniques, the number of times that one method is significantly better and worse (of 95 percent confidence level) than the others are assessed across experimented data sets. Let X; be the average value of validity index C 2 across n runs (n ¼ 50 in this evaluation) for a clustering method i 2 CM (CM is a set of 40 experimented clustering methods), on a specific data set 2 DT (DT is a set of six data sets). To obtain a fair comparison, this pair wise assessment is conducted on the results with six data sets, where the clustering results can be obtained for all the clustering methods. Also note that CM consists of five clustering algorithms for categorical data and 35 different cluster ensemble models, each of which is a unique combination of ensemble type (i.e., Type-I, Type- II(Fixed-k), Type-II(Random-k), Type-III(Fixed-k), and Type-III(Random-k)) and ensemble method (i.e., LCE, Base, CO+SL, CO+AL, CSPA, HGPA, and MCLA).

4.2.2. Bagging nearest neighbor classifiers

It is well known that the risk of a 1 nearest neighbor (1NN) classifier is at most twice the risk of the Bayes classifier, but there are no guarantees that this classifier will be consistent. By careful choice of the size of the resample's, bagging can lead to substantial improvements of the performance of the 1NN classifier. By taking a large number of re samples of the data of size n', the bagged nearest neighbor classifier will be consistent provided $n' \to \infty$ diverges but $n'/n \to 0$ as the sample size $n \to \infty$.

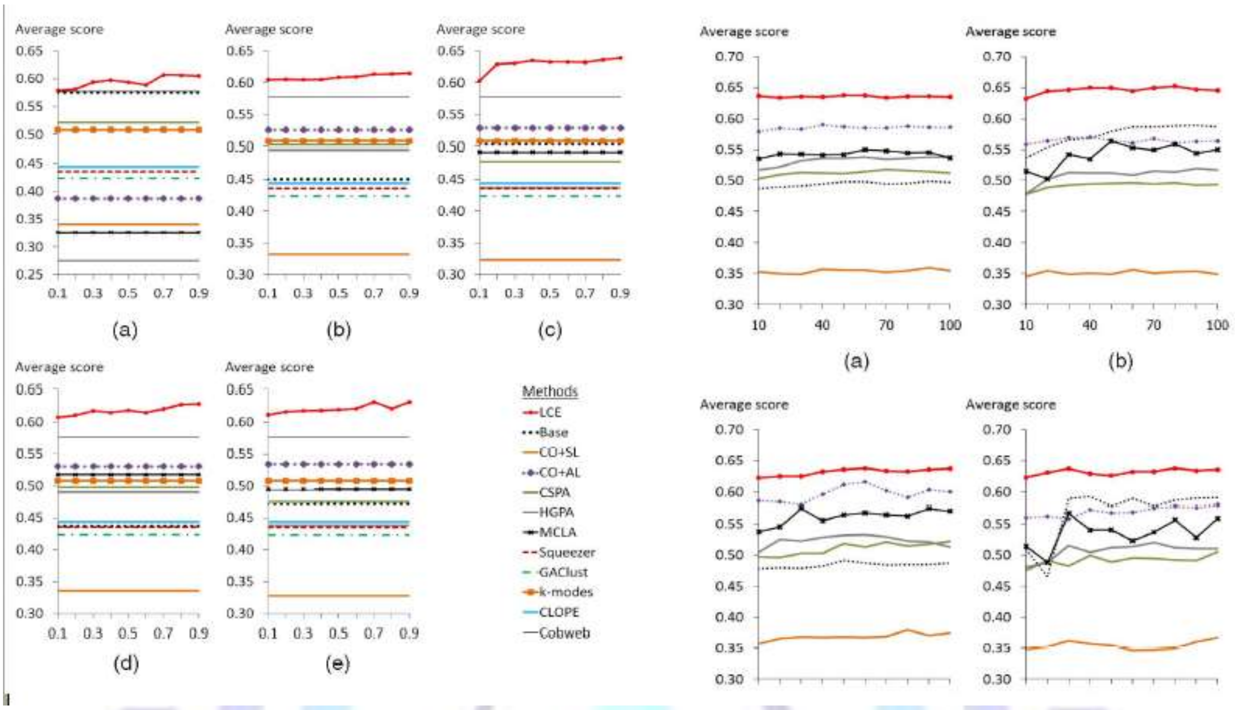
Under infinite simulation, the bagged nearest neighbor classifier can be viewed as a weighted nearest neighbor classifier. Suppose that the feature space is d dimensional and denote by $C^{bnn}_{n,n'}$ the bagged nearest neighbor classifier based on a training set of size n, with re samples of size n'. In the infinite sampling case, under certain regularity conditions on the class distributions, the excess risk has the following asymptotic expansion[2]

$$\mathcal{R}_{\mathcal{R}}(C^{bnn}_{n,n'}) - \mathcal{R}_{\mathcal{R}}(C^{Bayes}) = \left(B_1\frac{n'}{n} + B_2\frac{1}{(n')^{4/d}}\right)\{1+o(1)\},$$

for some constants B_1 and B_2. The optimal choice of n', that balances the two terms in the asymptotic expansion, is given by $n' = Bn^{d/(d+4)}$ for some constant B.

4.3 Parameter and Complexity Analysis

The parameter analysis aims to provide a practical means by which users can make the best use of the link-based framework. Essentially, the performance of the resulting technique is dependent on the decay factor (i.e., DC 2 ½0; 1), which is used in estimating the similarity among clusters and association degrees previously unknown in the original BM.



Graph: complexity Analysis based on link based Framework

We varied the value of this parameter from 0.1 through 0.9, in steps of 0.1, and obtained the results in Fig. 4. Note that the presented results are obtained with the ensemble size (M) of 10. The figure clearly shows that the results of LCE are robust across different ensemble types, and do not depend strongly on any particular value of DC. This makes it easy for users to obtain high-quality, reliable results, with the best outcomes being obtained with values of DC between 0.7 and 0.9. Although there is variation in response across the DC values, the performance of LCE is always better than any of the other clustering methods included in this assessment. Another important observation is that the effectiveness of the link-based measure decreases as DC becomes smaller. Intuitively, the significance of disclosed associations becomes trivial when DC is low. Hence, they may be overlooked by a consensus function and the quality of the resulting data partition is not improved.

Another parameter that may determine the quality of data partition generated by a cluster ensemble technique is the ensemble size (M). Intuitively, the larger an ensemble is, the better the performance becomes. Besides previous quality assessments, computational requirements of the link-based method are discussed here. Primarily, the time complexity of creating the RM is Np, where N is the number of data points. While P denotes the number of clusters in a Type-II or Type-III ensemble, it represents the cardinality of all categorical values in a direct ensemble (i.e., Type-I). Please consult Section III in the online supplementary for the details of the scalability evaluation.

5. CONCLUSIONS

This paper presents a novel, highly effective link-based cluster ensemble approach along with the Bootstrap aggregation technique to categorical data clustering for the better performance. It transforms the original categorical data matrix to an information-preserving numerical variation (RM), to which an effective graph partitioning technique can be directly applied. The problem of constructing the RM is efficiently resolved by the similarity among categorical labels (or clusters), using the Weighted Triple-Quality similarity algorithm.

REFERENCES:

[1] D.S. Hochbaum and D.B. Shmoys, “A Best Possible Heuristic for the K-Center Problem,” Math. of Operational Research, vol. 10, no. 2, pp. 180-184, 1985.

[2] L. Kaufman and P.J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis. Wiley Publishers, 1990.

[3] A.K. Jain and R.C. Dubes, Algorithms for Clustering. Prentice-Hall, 1998.

[4] P. Zhang, X. Wang, and P.X. Song, “Clustering Categorical Data Based on Distance Vectors,” The J. Am. Statistical Assoc., vol. 101, no. 473, pp. 355-367, 2006.

[5] J. Grambeier and A. Rudolph, “Techniques of Cluster Algorithms in Data Mining,” Data Mining and Knowledge Discovery, vol. 6, pp. 303-360, 2002.

[6] K.C. Gowda and E. Diday, “Symbolic Clustering Using a New Dissimilarity Measure,” Pattern Recognition, vol. 24, no. 6, pp. 567- 578, 1991.

[7] J.C. Gower, “A General Coefficient of Similarity and Some of Its Properties,” Biometrics, vol. 27, pp. 857-871, 1971.

[8] Z. Huang, “Extensions to the K-Means Algorithm for Clustering Large Data Sets with Categorical Values,” Data Mining and Knowledge Discovery, vol. 2, pp. 283-304, 1998.

[9] Z. He, X. Xu, and S. Deng, “Squeezer: An Efficient Algorithm for Clustering Categorical Data,” J. Computer Science and Technology, vol. 17, no. 5, pp. 611-624, 2002.

[10] P. Andritsos and V. Tzerpos, “Information-Theoretic Software Clustering,” IEEE Trans. Software Eng., vol. 31, no. 2, pp. 150-165, Feb. 2005.

[11] D. Cristofor and D. Simovici, “Finding Median Partitions Using Information-Theoretical-Based Genetic Algorithms,” J. Universal Computer Science, vol. 8, no. 2, pp. 153-172, 2002.

[12] D.H. Fisher, “Knowledge Acquisition via Incremental Conceptual Clustering,” Machine Learning, vol. 2, pp. 139-172, 1987.

[13] D. Gibson, J. Kleinberg, and P. Raghavan, “Clustering Categorical Data: An Approach Based on Dynamical Systems,” VLDB J., vol. 8, nos. 3-4, pp. 222-236, 2000.

[14] X.Z. Fern and C.E. Brodley, “Solving Cluster Ensemble Problems by Bipartite Graph Partitioning,” Proc. Int’l Conf. Machine Learning (ICML), pp. 36-43, 2004.

[15] A. Strehl and J. Ghosh, “Cluster Ensembles: A Knowledge Reuse Framework for Combining Multiple Partitions,” J. Machine Learning Research, vol. 3, pp. 583-617, 2002.

[16] H. Ayad and M. Kamel, “Finding Natural Clusters Using Multiclusterer Combiner Based on Shared Nearest Neighbors,” Proc. Int’l Workshop Multiple Classifier Systems, pp. 166-175, 2003.

[17] A.L.N. Fred and A.K. Jain, “Combining Multiple Clusterings Using Evidence Accumulation,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 27, no. 6, pp. 835-850, June 2005.

[18] S. Monti, P. Tamayo, J.P. Mesirov, and T.R. Golub, “Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data,” Machine Learning, vol. 52, nos. 1/2, pp. 91-118, 2003.

[19] N. Iam-On, T. Boongoen, and S. Garrett, “Refining Pair wise Similarity Matrix for Cluster Ensemble Problem with Cluster Relations,” Proc. Int’l Conf. Discovery Science, pp. 222-233, 2008.

[20] T. Boongoen, Q. Shen, and C. Price, “Disclosing False Identity through Hybrid Link Analysis,” Artificial Intelligence and Law, vol. 18, no. 1, pp. 77-102, 2010.

[21] L. Getoor and C.P. Diehl, “Link Mining: A Survey,” ACM SIGKDD Explorations Newsletter, vol. 7, no. 2, pp. 3-12, 2005.

[22] D. Liben-Nowell and J. Kleinberg, “The Link-Prediction Problem for Social Networks,” J. Am. Soc. for Information Science and Technology, vol. 58, no. 7, pp. 1019-1031, 2007.

[23] J. Kittler, M. Hatef, R. Duin, and J. Matas, “On Combining Classifiers,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 3, pp. 226-239, Mar. 1998.

[24] L.I. Kuncheva and D. Vetrov, “Evaluation of Stability of K-Means Cluster Ensembles with Respect to Random Initialization,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 28, no. 11, pp. 1798-1808, Nov. 2006.

[25] G. Karypis and V. Kumar, “Multilevel K-Way Partitioning Scheme for Irregular Graphs,” J. Parallel Distributed Computing, vol. 48, no. 1, pp. 96-129, 1998.

[26] A. Ng, M. Jordan, and Y. Weiss, “On Spectral Clustering: Analysis and an Algorithm,” Advances in Neural Information Processing Systems, vol. 14, pp. 849-856, 2001.

[27] M. Al-Razgan, C. Domeniconi, and D. Barbara, “Random Subspace Ensembles for Clustering Categorical Data,” Supervised and Unsupervised Ensemble Methods and Their Applications, pp. 31-48, Springer, 2008.

[28] Z. He, X. Xu, and S. Deng, “A Cluster Ensemble Method for Clustering Categorical Data,” Information Fusion, vol. 6, no. 2, pp. 143-151, 2005.

Author Profiles:

Mr. S. Pavan Kumar Reddy received his B.Tech (CSE) from Vaagdevi Institute of Technology and Sciences, JNTU-Anantapur, and pursuing M.Tech in Computer Science and Engineering from Vaagdevi Institute of Technology and Sciences, JNTU-Anantapur.

Mr.U.Sesadri received his M.Sc (CS) from SriVenkateswara University-Tirupati, M.Tech (CSE) from Satyabhama University. Working as HOD in CSE in Vaagdevi Institute of Technology and Sciences, under JNTU-Anantapur and have 10 years of experience.