Integrated Framework Using Frequent Pattern for Clustering Numeric and Nominal Data Sets

Aswathy Asok, T.J. Jisha, Sreeja Ashok and M.V. Judy

Abstract Clustering is an exploratory technique in data mining that aligns objects which have a maximum degree of similarity in the same group. The real-world data are usually mixed in nature, i.e., it can contain both numeric and nominal data. Performance degradation is a major challenge in existing mixed data clustering due to multiple iterations and increased complexities. We propose an integrated framework using frequent pattern analysis, frequent pattern-based framework for mixed data clustering (FPMC) algorithm, to cluster mixed data in a competent way by performing a one-time clustering along with attribute reduction. This algorithm comes under divide-and-conquer paradigm, with three phases, namely crack, transformation, and merging. The results are promising when the algorithm is applied on benchmark datasets.

Keywords Frequent pattern analysis · Clustering · Normalization · Sum of squared error · FPMC

1 Introduction

Clustering is the process of identifying the classes of objects with similar characteristics. Data clustering segregates the similarities and variances in the database to form groups of related data as either classes or clusters. Apart from classification,

A. Asok (✉) · T.J. Jisha · S. Ashok · M.V. Judy
Department of Computer Science and IT, Amrita School of Arts and Sciences,
Amrita Vishwa Vidyapeetham, Kochi, India
e-mail: achu2061991@gmail.com

T.J. Jisha
e-mail: jishatj13@gmail.com

S. Ashok
e-mail: sreeja.ashok@gmail.com

M.V. Judy
e-mail: judy.nair@gmail.com

S.C. Satapathy et al. (eds.), *Proceedings of International Conference on ICT for Sustainable Development*, Advances in Intelligent Systems and Computing 408, DOI 10.1007/978-981-10-0129-1_55

clustering is an un-supervised learning method to uncover the causal structures and patterns of a given dataset. It is also known as automatic classification in the sense, and data objects can be treated as an implicit class. The distinct advantage is that it can automatically find the groupings. The clustering methods are mainly divided into the following categories: partitioning, hierarchical, density-based, and grid-based methods. Partitioning method is a popular heuristic method which improves the segregation by moving the objects from one group to another by a local optimum approach. k-means and k-medoids are most commonly used partitioning methods. Hierarchical method breaks down the data objects to various levels of hierarchies. This method has two approaches, agglomerative and divisive. The agglomerative approach builds the hierarchies in a bottom-up fashion, whereas divisive approach does the same in the top down. Density-based method solves the difficulty in finding arbitrary-shaped clusters. The clusters are grown on the basis of density to solve the issue. The high density area is termed as clusters, whereas the sparse areas are used to differentiate the clusters. Grid-based method is one of the high-speed clustering methods, which divides the object space into a number of cells that form a grid structure. The processing time depends upon cells in each dimension of the quantized space [1]. Most clustering algorithms focus on numerical data clustering. However, real data sets are primarily mixed in nature which contains both numerical and categorical data types. Major challenge in mixed data clustering is to find a single clustering solution for both data types with improved performance and reduced complexities.

In this paper, we propose an efficient clustering algorithm for mixed data, FPMC which performs clustering after crack, transformation, and merge phase. In crack phase, the total data set is divided into nominal and numerical packs. In transformation phase, frequent patterns are mined to extract frequency-token which are numerical substitutes to nominal values. Attribute reduction is achieved by converting 'n' number of nominal attributes to a single numerical attribute. Merge phase combines the output of transformation phase and numerical attributes which will undergo normalization before any numerical clustering algorithm is applied.

Section 2 deals with the existing algorithms in mixed data clustering and explains the advantages and disadvantages of each method. Section 3 talks about the proposed work in detail, the process flow along with implementation details and the validation measures used in the work. Section 4 gives the experiment setup and the accuracy of the FPMC algorithm. Section 5 presents the conclusion and the extension of the proposed framework.

2 Related Works

Frequent pattern analysis has been a very interesting and focused area of research in data mining. The frequent pattern analysis finds item sets, subsequences, or substructure that appears frequently in a dataset. The frequency is measured in terms of number of occurrences of a particular sequence in the dataset with two important

matrices, support and confidence. Support means the probability of occurrence, whereas confidence is the certainty of occurrence of an event. In frequent pattern analysis, the frequency greater than or equal to the minimum support is checked. For example, following are the patterns obtained of an application:

$$\begin{aligned} &I1, I2, I3 = \ > 2 \\ &I3, I4, I5 = \ > 3 \\ &I2, I3, I5 = \ > 3 \\ &I4, I5, I1 = \ > 2 \\ &I1, I3, I5 = \ > 4 \end{aligned}$$

If the minimum support is set at 3, then the frequent patterns obtained are {{I1, I2, I3}, {I4, I5, I1}, and {I1, I3, I5}}. These are the only patterns that meet the general criteria, i.e., select the only pattern that has support count greater than or equal to minimum support count. The efficiency of association rule generation is different for the various types of frequent item set, namely Eclat, Apriori, and FP growth algorithms.

Eclat algorithm uses vertical database layout where each item of transaction is stored along with its cover in the database. This computes the support by interaction-based approach. The disadvantage is that it is suitable only for small data sets [2]. Apriori algorithm is based on level-wise search. This algorithm begins with the selection of one item and then proceeds by adding the item one at a time and is checked against the support which is required as per the requirement of the application [3]. The FP growth tree is the approach in which the problem of Apriori algorithm is solved by introducing a compact data structure which avoids the need of candidate generation. The execution time of this algorithm is large due to complex data structure and also it is difficult to fit in the main memory [4].

Commonly used algorithms for mixed data are Ralambondrainy [5], k-prototype [6], CLARA [7], and cluster ensemble [8]. Ralambondrainy proposed a new algorithm for mixed data set by converting the categorical attributes into binary and treating the binary as numerical value. By the conversion into a numerical value, k-means algorithm can be directly applied. The drawback of this approach is the space costs and computational complexity with increase in binary attributes in a data set. The other drawback is with 'mean', which do not give the cluster character. The k-prototype algorithm uses squared Euclidean distance as the dissimilarity measure for numerical (Sr) and the number of mismatches between two objects (Sc) for categorical. Sr+∞Sc is the combined dissimilarity measure where ∞ is the weightage given to provide equal importance to both sides, i.e., numeric and nominal attributes. It uses k-modes to update categorical attribute and all other procedures are similar to k-means. As k-prototype is derived from traditional k-means, it is having the same problems that k-means possess. CLARA combines sampling along with clustering program. This method finds objects by k-medoids, so it clusters categorical attributes. This is inefficient for large data set clustering. Cluster ensemble approach splits the data set into nominal and numerical and then

applies clustering separately and at last combines the results of clustering. After combining the result, numerical or nominal clustering method is applied. Even though it avoids the problems with other clustering approaches, this approach has got high clustering complexity. The error rate will get multiplied as clustering is conducted three times.

3 Proposed System

Frequent pattern-based framework for mixed data clustering (FPMC) algorithm addresses the major challenges in the existing clustering solutions like performance degradation and space computational complexities by avoiding repeated iterations and optimizing using attribute reduction. Nominal attributes are replaced with the count of frequent patterns that are mined from the data set. For this FPMC uses Apriori due to its easiness in implementation and simplicity in data structure. Apriori is also suitable for large data sets.

The process flow of FPMC algorithm is given in Fig. 1.

First phase of the algorithm is crack stage where algorithm first separates the total data set into nominal and numerical after preprocessing. The crack stage produces two results, one is nominal and other numerical pack. The nominal pack undergoes the second phase, i.e., transformation phase where frequent pattern analysis is done on the dataset to obtain a frequency-token value for each frequent

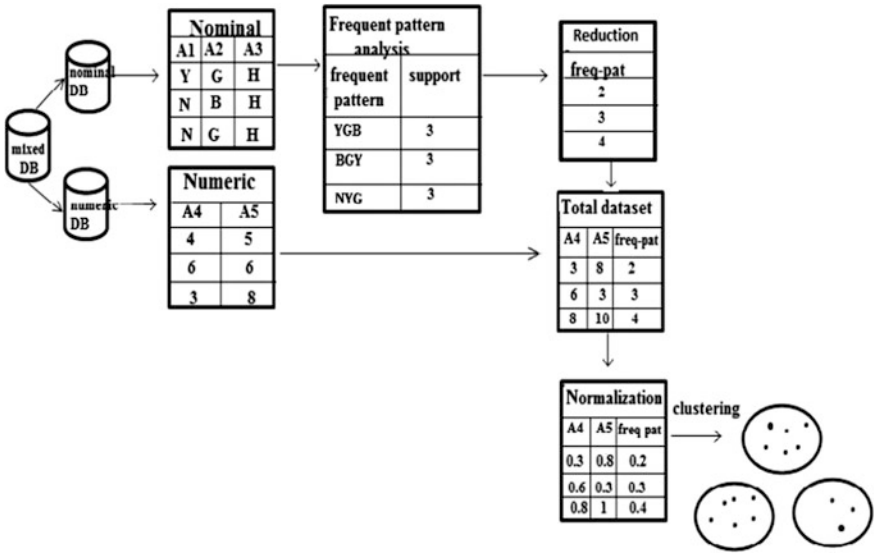

Fig. 1 FPMC flow of execution

item set. This value is obtained by analyzing the nominal pack with the frequent patterns derived from Apriori analysis by evaluating Eq. (1).

Let P_1, $P_{2....}$ P_n be the frequent patterns obtained after Apriori analysis:

$$\mathrm{RS}_{\mathrm{value}} = \begin{Bmatrix} P_i, & \mathrm{Count}++;\mathrm{flag} = \mathrm{Valid} \\ !P_i, & \mathrm{flag} = \mathrm{Invalid} \end{Bmatrix} \tag{1}$$

where P_i represents the ith frequent pattern and $\mathrm{RS}_{\mathrm{value}}$ is the result of row-wise scan. If a match is found, i.e., $\mathrm{RS}_{\mathrm{value}} = P_i$, increment the count and mark the row as valid; else mark it as invalid.

Third step is the merging phase, where numeric and frequency-token attributes are merged forming a complete numeric dataset. Normalization is done on the dataset for variance stabilization. Normalization is a process in which all attributes are given an equal weight. This is particularly useful for distance measures while used in clustering. There are mainly three types of normalization techniques available namely min-max, z-score, and decimal scaling normalization. Min-max normalization refers to the process of altering the original data into a specified range in a linear fashion. For mapping a v value, of an attribute A from range [min_A, max_A] to a new range [$\mathrm{new\_min}_A$, $\mathrm{new\_max}_A$], the computation is given by Eq. (2):

$$\frac{v - \mathrm{min}_A}{\mathrm{max}_A - \mathrm{min}_A}(\mathrm{new\_max}_A - \mathrm{new\_min}_A) + \mathrm{new\_min}_A \tag{2}$$

where 'v' is the new value in the required range.

Z-score normalization is based on mean and median, and it is also called as zero mean normalization. The formula is given in Eq. (3):

$$d^* = \frac{d - \mathrm{mean}(P)}{\mathrm{std}(P)} \tag{3}$$

where mean(p) is the sum of the all attribute values of P and Std(P) is the standard deviation of all values of P. Decimal scale normalization is based on the decimal point movement depending on the absolute values of the attributes. The formula is given below in Eq. (4):

$$\max(|\mathrm{d}|) < 1.[5] \tag{4}$$

Z-score normalization is used in the FPMC algorithm as it maintains the range and dispersion of the data set, i.e., Standard deviation/variance. After normalization an efficient numerical clustering algorithm is applied. Pseudocode of FPMC is given in Table 1.

Table 1 FPMC Algorithm

Step 1 : Partitioning
After the replacement of missing values from the dataset it is divided into numerical and nominal packs.
Step 2 : Transformation.
Step 2.1: Frequent patterns are generated using Apriori.
Step 2.2: Perform row wise scan on the nominal attributes and execute the operations in *Equation* (1) for all instances. After the row wise scanning of entire dataset, we get the value for frequency-token. For the first frequency pattern i.e. P_1 go to Step 2.3 otherwise Step 2.4.
Step 2.3: Set frequency-token attribute as count value for all rows marked as valid. And also store a copy of count to init-token.
Step 2.4: If (init-token < count)
Set frequency-token as count value if it is empty, or replace it with count value if it is non-empty.
Step 3 : Merging
Join the numerical and frequency-token attribute.
Step 4 : Normalization
Z-score normalization is used in FPMC algorithm as it maintains the range and dispersion of the data set
Step 5 : Clustering
Perform any numerical clustering algorithm.
Step 6 : Validation of the results
SSE is used as evaluation criteria for FPMC algorithm. The Sum of squared error for each data point is the distance to the nearest cluster. The clustering produces good results with small value for SSE with minimum number of clusters. Equation (5) gives the formula for SSE calculation where m_i represent the mean of the cluster and x the data point C the cluster.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x) \quad (5)$$

4 Experiments and Results

To evaluate the effectiveness of FPMC algorithm, three different types of real-time data sets were taken into account. The accuracy of the experiment is evaluated using sum of squared error and percentage of incorrectly clustered instances. The experiments consider two algorithms for the analysis. They are simple k-means and cobweb. In simple k-means SSE for a data set without class label and percentage of incorrectly clustered instances for a data set with class label is used. The cobweb algorithm is used to show the percentage of incorrectly clustered instances of data set with class label. We have taken three data sets of different characteristics, for better analysis. They are automobile, labor, and post-operative patient datasets.

Automobile dataset has eighteen numeric and six nominal attributes. The aim of choosing this data set was to cluster mixed data containing equal number of numerical and nominal attributes. Post-operative patient dataset contains one numeric and eight nominal attribute. In this data set, 64 instances belonging to the patients are sent to the general hospital floor; 24 instances represent patients prepared to go home and two instances of patients sent to the intensive care unit. Labor dataset contains eight numeric and eight nominal attributes. Table 2 shows the comparison results of SSE for the data sets with and without class labels. FPMC is compared with simple k-means. We cannot make use of SSE validation in cobweb because it is not based on distance measures.

Table 2 Comparison results of SSE

Number of clusters	FPMC			k-means		
	3	4	5	3	4	5
Labor	13.48	12.78	10.89	119.52	106.30	99.23
Automobile	68.71	55.41	46.79	607.29	560.59	555.31
Post-operative patient	19.81	18.75	5.02	178.69	169.84	150.78

Table 3 Comparison results of percentage of incorrectly clustered instances

Number of clusters	FPMC			k-means			Cobweb
	3	4	5	3	4	5	
Labor	36.73	38.77	42.85	36.84	50.87	54.38	85.55
Post-operative patient	46.47	40.84	43.66	52.22	63.33	71.11	52.63

Table 3 gives the comparison results of the FPMC with simple k-means and cobweb using percentage of incorrectly clustered instances as a validation measure. This validation technique is applicable only for data sets with class label.

5 Conclusions and Future Work

The main objective of clustering is to group similar instances of a data set. The grouping of instances is made on the basis of similarity measures. Even though there are many distance measures available, most of them are applied either on numeric or nominal data. But the real-world data are usually mixed in nature. So we cannot directly apply these distance measures. For this most algorithms for mixed data require partitioning of a dataset into nominal and numeric which increases the complexity and degrades the clustering result. In our proposed work we try to find a solution to this problem by transforming nominal data into numeric. The future plan is to improve the performance using efficient pattern generation and clustering algorithms in multidimensional dataset.

Acknowledgments This work is supported by the DST Funded Project, (SR/CSI/81/2011) under Cognitive Science Research Initiative in the Department of Computer Science, Amrita School of Arts and Sciences, Amrita Vishwa Vidyapeetham University, Kochi.

References

1. Han, J., Kamber, M., & Pei, J. (2012). *Data mining: Concepts and techniques*. Morgan Kaufmann, USA.
2. Hipp, J., Myka, A., Wirth, R., & Güntzer, U. (1998). *A new algorithm for faster mining of generalized association rules*

3. Liu, B., Ma, Y., & Wong, C. K. (2002). Improving an association rule based classifier. *Principles of Data Mining and Knowledge Discovery Lecture Notes in Computer Science*, *(1910, 2000)*, 504–509.
4. Wang, K., Tang, L., Han, J., & Liu, J. (2002). Top down FP-growth for association rule mining. *Advances in Knowledge Discovery and Data Mining Lecture Notes in Computer Science*, *2336*, 334–340
5. Huan, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, *2*, 283–304.
6. Huang, Z. (1997). Clustering large data sets with mixed numeric and categorical values. In *Asia Conference of Knowledge Discovery and Data*.
7. Ahmad, A., & Dey, L. (2007). *A k-mean clustering algorithm for mixed numeric and categorical Data*.
8. He, Z., Xu, X., & Deng, S. (2005). *Clustering mixed numeric and categorical data: A cluster ensemble approach*.