Scaling Clustering Algorithm for Data with Categorical Attributes.

ERENDIRA RENDÓN LARA, R. BARANDELA
Computer Systems Department
Technologic Institute of Toluca, México
Av. Ins. Tecnológico s/n, Ex. La virgen, Metepec, Edo. de Méx.
MÉXICO

Abstract: Clustering constitutes an important task inside the fields of Pattern Recognition and Data Mining. Clustering of categorical data is a difficult problem and has not received the attention its importance deserves. In the present paper, we introduce a new clustering method to work with categorical data. The algorithm is easily scalable and yields better clustering results that the well-known K-MODES and Rock algorithms.

Key-words: Scaling clustering, Categorical attributes, connected component, Composite Object.

1 Introduction
Clustering is an effective technique for exploratory data analysis and has been studied for several years. It has found applications in a wide variety of areas such as pattern recognition, statistical data analysis and modeling, data mining, fraud detection, marketing, and other business applications. The basic clustering problem consists of grouping a data set into subsets (i.e. clusters), such that items in the same subset are similar to each other, whereas items in different subsets are as dissimilar as possible. The basic idea is to uncover a structure that is already present in the data. Most of the existing clustering algorithms can be classified into two main categories: hierarchical and partitional.

Partitional clustering algorithms attempt to generate a partition of the data set that optimizes a certain criterion function. In these algorithms, each cluster is represented by a prototype or representative object (e.g., the mean or centroid), and the sum of the distances from each data item to its nearest prototype is usually employed as the criterion function.

Hierarchical algorithms do not attempt to construct a single partition with $k$ clusters. Instead of that, they are concerned with all values of $k$ in the same run. These clustering procedures yield a nested sequence of partitions that corresponds to a graphical representation known as the dendrogram (an inverted tree diagram). Hierarchical procedures can be either agglomerative or divisive.

In Pattern Recognition and Data Mining practical applications it is frequently required to deal with high volumes of data (thousands or millions of records with tens or hundreds of attributes). This characteristic excludes the possibility of using many of the traditional clustering algorithms. Besides, this kind of applications is often done with data containing categorical attributes.

Clustering algorithms require a large amount of computations of distances among objects and centers of clusters. Hence, their complexity is dominated by the number of objects. On the other hand, there is an explosive growth of business or scientific databases storing huge volumes of data. One of the main challenges of today’s data mining systems is their ability to scale up to very large data sets. However, there are applications where the entire data set cannot be stored in the main memory because of its size. There are currently three possible approaches to solve this problem.

The objects set can be stored in a secondary memory and subsets of this data clustered independently, followed by a merging step to yield a clustering of the entire database. This approach is called, the scale up approach. In the approach incremental, the entire database is stored in a secondary memory and the objects are transferred to the main memory one at a time for clustering. Only the cluster representations are stored in the main memory to alleviate the space limitations, the BIRCH algorithm has a pre-clustering step for carries out a summary of the entire database.
Another approach requires the transformation of the clustering algorithm to an optimized parallel one for a specific architecture. There are several parallel clustering algorithms that are proposed in the literature, both for partitional clustering and for hierarchical clustering. Recently, the problems of clustering categorical data and scalability clustering started receiving interest [3-6]. This paper presents a clustering algorithm: SCCA (Scaling Clustering for Categorical Data), which has been designed to handle large databases and to work with categorical data.

2 Definitions

2.1 Categorical Object
An event is a pair relating its own features and values. It is denoted by $E = (i, j)$, where the feature $iX$ takes the values of $iE$ and $UE = X_i \subseteq U_i$. $U_i$ is the subset of values that the feature $iX$ takes. $Ui$ is a non-arranged subset of every possible value that $iX$ may take (domain of $iX$). Example of event:

$E = (Eye\_color = \{green, blue, red\}) \land (Hair\_color = \{black, brown\}) \land (Blood\_type = \{B+, A+\})$

Here, this categorical object has the following features:
1. Eyes\_color is green, blue or red.
2. Hair\_color is black or brown.
3. Blood\_type is B+ or A+.

2.2 Intersection between categorical objects
Let objects $X_i = [X_{i1} = E_{i1}] \land [X_{i2} = E_{i2}] \land \ldots \land [X_{id} = E_{id}]$ and $X_j = [X_{j1} = E_{j1}] \land [X_{j2} = E_{j2}] \land \ldots \land [X_{jd} = E_{jd}]$ be a pair of objects of $U^{(d)}$. Then, the intersection between $X_i$ and $X_j$ is defined by:

$$X_i \cap X_j = [E_{i1} \land E_{j1}] \land [E_{i2} \land E_{j2}] \land \ldots \land [E_{id} \land E_{jd}]$$

where $X_i \cap X_j$ is the intersection of the $d$-th values of $X_i$ and $X_j$.

2.3 Union between categorical objects
Let objects be a pair of objects $X_i = [X_{i1} = E_{i1}] \land [X_{i2} = E_{i2}] \land \ldots \land [X_{id} = E_{id}]$ and $X_j = [X_{j1} = E_{j1}] \land [X_{j2} = E_{j2}] \land \ldots \land [X_{jd} = E_{jd}]$ of $U^{(d)}$. Then, the union $X_i$ and $X_j$ is defined by:

$$X_i \cup X_j = [E_{i1} \lor E_{j1}] \land [E_{i2} \lor E_{j2}] \land \ldots \land [E_{id} \lor E_{jd}]$$

where $X_i \cup X_j$ is the union of the $d$-th value of $X_i$ and $X_j$ and is defined as the union of $X_i$ and $X_j$.

2.4 Similarity between categorical objects
We used a similar concept to Ichino [8], to define our similarity measure. The distance between objects $X_i = [X_{i1} = E_{i1}] \land [X_{i2} = E_{i2}] \land \ldots \land [X_{id} = E_{id}]$ and $X_j = [X_{j1} = E_{j1}] \land [X_{j2} = E_{j2}] \land \ldots \land [X_{jd} = E_{jd}]$ in $U^{(d)}$ is calculated by:

$$d_p(X_i, X_j) = \left( \sum_{k=1}^{d} C_k \psi(E_{ik}, E_{jk})^p \right)^{1/p}, \quad p \geq 1$$

where:

$$\psi(E_{ik}, E_{jk}) = \frac{\phi(E_{ik}, E_{jk})}{U_k}, \quad k = 1 \ldots d$$

$U_k$ is the number of possible values included in the domain $U_k$ and

$$\phi(E_{ik}, E_{jk}) = \left| \frac{E_{ik} \cup E_{jk} \setminus [E_{ik} \cap E_{jk}]}{U_k} \right|, \quad k = 1 \ldots d$$

$C_k$ is a weighting coefficient, to control the relative importance of $E_k$ or $C_k = \frac{1}{d}$ when all the $E_k$ events have the same weight, for $C_k, k = 1 \ldots d$ and $C_k > 0$, then, this distance satisfies $0 \leq d_p(X_i, X_j) \leq 1$. We transform the Eq.6 into similarity [9], such as:

$$\delta(X_i, X_j) = 1 - d_p$$
2.5 Composite Object
A composite object (CO) is a new object resulted from the combination of two or more objects. Let objects
\[ X_i = [x_{i,1} = e_{i,1}] \land [x_{i,2} = e_{i,2}] \land \ldots \land [x_{i,d} = e_{i,d}] \]
and
\[ X_j = [x_{j,1} = e_{j,1}] \land [x_{j,2} = e_{j,2}] \land \ldots \land [x_{j,d} = e_{j,d}] \]
be of \( U^{(d)} \); then, a composite object is the result of combining \( X_i \) and \( X_j \), which is calculated as follows:
\[ CO = X_i \cup X_j = [e_{i,1} \cup e_{j,1}] \land [e_{i,2} \cup e_{j,2}] \land [e_{i,d} \cup e_{j,d}] \quad (10) \]

2.6 \( \beta \)-connected component
Let \( CD = \{X_1, X_2, \ldots, X_n\} \) be the group of categorical objects in \( U^{(d)} \) for \( i = 1, \ldots, n \), all of them described in \( X_i = [x_{i,1} = e_{i,1}] \land [x_{i,2} = e_{i,2}] \land \ldots \land [x_{i,d} = e_{i,d}] \)

**Definition 1:** Let objects
\[ X_i = [x_{i,1} = e_{i,1}] \land [x_{i,2} = e_{i,2}] \land \ldots \land [x_{i,d} = e_{i,d}] \]
and
\[ X_j = [x_{j,1} = e_{j,1}] \land [x_{j,2} = e_{j,2}] \land \ldots \land [x_{j,d} = e_{j,d}] \]
be in \( U^{(d)} \) two descriptions of categorical objects and \( \beta \in [0,1] \) a similarity threshold. It is considered that objects \( X_i \) and \( X_j \) are \( \beta \)-Similar, if and only if \( S(X_i, X_j) \geq \beta \).

**Definition 2:**
Let \( C \subseteq CD, C \neq \emptyset \), be a \( \beta \)-connected component if and only if:
1. \( \forall X_i, X_j \in C \exists X_{i_k}, X_{j_q} \subseteq C \). \( X_i = X_{i_k} \land X_j = X_{j_q} \land \forall p \in [1, K, q - 1] \):
   \[ S(X_{p}, X_{q}) \geq \beta \]. This condition indicates that, for any pair of objects of \( C \), there is a succession of elements in \( C \), starting in \( X_i \) and ending in \( X_j \), so that each one is \( \beta \)-Similar to the next.
2. \( \forall X_i \in CD \left[ X_j \in C, S(X_i, X_j) \geq \beta \Rightarrow X_i \in C \right] \). This condition establishes that outside \( C \) there is no object \( \beta \)-Similar to the objects of \( C \).
3. When a connected component has an object, it is considered a degenerated \( \beta \)-connected component.

2.7 Example
Table 1 shows the similarity matrix of 5 objects. Let \( \beta = .8 \) be; applying definition 2, we obtain the connected components following: \( C_1 = \{ X_1, X_2 \} \) and \( C_1 = \{ X_3, X_4, X_5 \} \)

<table>
<thead>
<tr>
<th>CD</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>X5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>.0</td>
<td>.8</td>
<td>.5</td>
<td>.6</td>
<td>.7</td>
</tr>
<tr>
<td>X2</td>
<td>.8</td>
<td>.0</td>
<td>.7</td>
<td>.6</td>
<td>.5</td>
</tr>
<tr>
<td>X3</td>
<td>.5</td>
<td>.7</td>
<td>.0</td>
<td>.7</td>
<td>.9</td>
</tr>
<tr>
<td>X4</td>
<td>.6</td>
<td>.6</td>
<td>.7</td>
<td>.0</td>
<td>.8</td>
</tr>
<tr>
<td>X5</td>
<td>.7</td>
<td>.5</td>
<td>.9</td>
<td>.8</td>
<td>.0</td>
</tr>
</tbody>
</table>

3 The SCCA Clustering Algorithm
This paper introduces a new procedure: SCCA (Scaling Clustering for Categorical Data). It is a clustering algorithm designed to work with data described with categorical attributes. One of its main advantages is that it can handle databases of any size, making a summary of the data. Besides, the algorithm does not require knowing beforehand the number of groups to be formed. The main task of SCCA is to summarize the entire database. To do this, the database is processed in blocks to obtain, from each block, composite objects which will be the representatives of each one of the formed groups. SCCA consists of two phases: summarizing and labeling, see Fig. 1. The summary of the entire database is obtained with an iterative procedure. In the labeling phase, each object in the database receives the label of its nearest representative or prototype. The summary process consists of finding prototypes.

The input file is read block by block and the size of each block is given by the size of the main memory available. The summary process is applicable to each block and the obtained results (composite objects) are saved to disk (Output_File), when the whole input file has been read, the prototypes are in the Output_File. After that, the summary process is run with the Output_File to obtain the prototypes of the founded clusters.

3.1 Obtain prototypes "Summary"
1. Read Block
2. Calculate Similarity Matrix, with the Ec. 9
3. Calculate Connected Components \( (C) \) with the Def. 2
4. Calculate Composite Objects of $C$ with the Eq. 10

3.2 Labeling
In this phase a class label is placed to the data set. The objects of the database are labeled with the nearest prototype using the nearest neighbor criterion.

4 Experimental results
To assess the adequacy of SCCA, experiments with both real and synthetic datasets were performed. The real data was used to evaluate the clustering quality of SCCA. We used the number of misclassified objects as a measure of quality of clustering. We also did a comparison with the groups generated by the K-modes algorithm; this comparison was carried out with the K-Modes algorithm because it is one of the most popular in the community of data mining. The synthetic datasets were used to demonstrate the scalability of SCCA.

4.1 Real Datasets
The real data used in the experiments were taken from the repository of the University of California at Irvine [http://www.ics.uci.edu/Mlearn/MLRepository.html]. In all cases, the attributes are categorical. The description of these data is shown in Table 2.

Table 2. Characteristics of the data sets

<table>
<thead>
<tr>
<th>Data Sets</th>
<th>Number of Records</th>
<th>Number of Attributes</th>
<th>Number of classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mushroom</td>
<td>8124</td>
<td>22</td>
<td>2</td>
</tr>
<tr>
<td>Connect-4</td>
<td>67557</td>
<td>42</td>
<td>3</td>
</tr>
<tr>
<td>Kr-vs-Kp</td>
<td>3196</td>
<td>36</td>
<td>2</td>
</tr>
<tr>
<td>Tic-tac-toe</td>
<td>958</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Congressional votes</td>
<td>435</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

4.2 Results with real-life datasets
Table 3 contains the result of running the SCCA, K-Modes[6] and Rock[4] algorithms with the Mushroom, Kr-vs-kp and Connect-4 datasets.

Table 3. Comparison of the clustering quality (Misclassified objects (%))

<table>
<thead>
<tr>
<th>Data Sets</th>
<th>SCCA</th>
<th>K-Modes</th>
<th>Rock</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mushroom</td>
<td>6.08</td>
<td>7.42</td>
<td>19.68</td>
</tr>
<tr>
<td>Kr-vs-Kp</td>
<td>44.19</td>
<td>45.03</td>
<td>53.08</td>
</tr>
<tr>
<td>Connect-4</td>
<td>33.17</td>
<td>34.51</td>
<td>40.32</td>
</tr>
</tbody>
</table>

Mushroom: The K-Modes algorithm formed 20 clusters, of those which 11 are pure clusters and with an average of 7.42 % misclassified objects. The SCCA algorithm was run with $\beta$=0.7 forming 20 clusters and 6.08% of misclassified objects. The Rock algorithm was run with $k=8, \theta=.90$ and $S=500$.

Kr-vs-Kp: The K-Modes algorithm formed 6 clusters, with 45.03% the misclassified objects. The SCCA algorithm was run with $\beta=0.5$ forming 6 clusters, with an average of 44.19 % the misclassified objects. Rock algorithm was run with $k=2, \theta=.5$ and $S=500$.

Connect-4: The K-Modes algorithm formed 4 clusters, an average of 34.51% the misclassified objects. The SCCA algorithm as run with $\beta=0.5$
forming 4 clusters, with an average of 33.17% the misclassified objects. Rock algorithm was run with \( k=4, \theta = .70 \) and \( S=2500 \).

To prove the scalability of algorithm SCCA, we worked with data set Connect-4. For this purpose, we randomly formed six data sets of sizes 10000, 20000, 30000, 40000, 50000 and 60000, respectively. Different values of \( \beta (0.6, 0.5 \) and 0.4) were employed. Figure 2 presents a graphic with the results obtained by SCCA, considering the time of execution and size of the dataset. The time of execution is variable for the same dataset, depending on the value of \( \beta \). That is, for higher values of \( \beta \), a greater number of iterations are needed because each iteration performs fewer combinations to create the composite objects.

The purpose of this experiment was to test the quality of clustering and the run time, when SCCA works with the entire dataset and when SCCA is execute in blocks.

The Mushroom, Tic-Tac-Toe, Congressional Votes and kr-vs-kp datasets were executed with SCCA, assumption that all objects to be clustered can reside in memory at the same time. For comparison purposes, we also run SCCA with the same datasets in blocks, the result are presented in the table 4 and 5.

The purpose of this experiment was to test the quality of clustering and the run time, when SCCA works with the entire dataset and when SCCA is execute in blocks.

Tables 4 and 5 shows the relationships between the quality clustering the SCCA when was executed with the entire Mushroom, Tic-Tac-Toe, Congressional Votes and kr-vs-kp datasets. We used this datasets for this test, because they fit in memory. For example the quality of clustering of the mushroom (entire) dataset was 4.2% the misclassified objects against 5.63% the misclassified objects when SCCA was executed in blocks. The running time is less when SCCA is run in blocks that when is run with the entire dataset.

These results indicate that we can use the SCCA algorithm without losing quality of clustering with a smaller execution time.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Execution Time (Sec)</th>
<th>(% ) of misclassified objects</th>
<th>( \beta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mushroom</td>
<td>1948</td>
<td>12</td>
<td>.7</td>
</tr>
<tr>
<td>Kr-vs-Kp</td>
<td>384</td>
<td>29.85</td>
<td>.5</td>
</tr>
<tr>
<td>Tic-tac-toe</td>
<td>44</td>
<td>34.24</td>
<td>.45</td>
</tr>
<tr>
<td>Congressional votes</td>
<td>16</td>
<td>19.06</td>
<td>.50</td>
</tr>
</tbody>
</table>

Table 4. Clustering obtained with entire datasets

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Execution Time (Sec)</th>
<th>(% ) of misclassified objects</th>
<th>( \beta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mushroom</td>
<td>1218</td>
<td>5.63</td>
<td>.7</td>
</tr>
<tr>
<td>Kr-vs-Kp</td>
<td>244</td>
<td>30.46</td>
<td>.5</td>
</tr>
<tr>
<td>Tic-tac-toe</td>
<td>10</td>
<td>35.72</td>
<td>.45</td>
</tr>
<tr>
<td>Congressional votes</td>
<td>3</td>
<td>13.42</td>
<td>.50</td>
</tr>
</tbody>
</table>

Table 5. Clustering obtained with the datasets in blocks

<table>
<thead>
<tr>
<th>Data set</th>
<th>Size Block (Kbytes)</th>
<th>( \beta )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100</td>
<td>250</td>
</tr>
<tr>
<td>Kr-vs-Kp</td>
<td>20.75</td>
<td>27.8</td>
</tr>
<tr>
<td>Connect-4</td>
<td>50.07</td>
<td>41.80</td>
</tr>
<tr>
<td>Mushroom</td>
<td>5.3</td>
<td>5.8</td>
</tr>
</tbody>
</table>

Table 6. Effects of memory size

In this paper we proposed a new scaling clustering algorithm for categorical data. The algorithm does not require the number of clusters to create. It can also works with large datasets. We purpose, a technique that consists in processing the database by
blocks. This clustering algorithm is performed iteratively until that the procedure obtains the summary of the database, represented by the composed objects (the representatives or prototypes). Afterward is carried out a labeling phase, where each one of the objects in the database receives the label of its nearest representative or prototype.

The results of our experimental study with database are very encouraging, as they demonstrate that SCCA not only outperforms K-Modes but also scales well for large databases without sacrificing clustering quality.

In future, we intend to test the SCCA algorithm with other similarity measure, and to change the clustering approach.

References:


