Proceedings of the 5th WSEAS International Conference on Telecommunications and Informatics, Istanbul, Turkey, May 27-29, 2006 (pp260-266)

How clustering can be useful to supermarkets?

Henda BELAID AJROUD[1], Hamida AMDOUNI[2]
[1]Département d'informatique
Institut Supérieur de Gestion de Tunis
Bouchoucha, Tunisie
[2]Départment d'informatique
Faculté des Sciences de Tunis
Campus du Belvédère, Tunisie

Abstract:
The uncertainty of the economic and social environment of the nowadays companies as well as the evolution of the relation with the customers have made necessary the analysis of the collected data in order to extract the knowledge that helps to take efficient decisions and to improve the profitability. In this domain, several tools and techniques have been used. In this paper, we present an incremental clustering approach to be used for the analysis of supermarkets data.

Key-Words: Customers, Properties, Transactions, Incremental, Clustering, Rectangle

1 Introduction

With the apparition of new given data such as the globalization and the internationalization of exchanges, the economical and the social environment of the companies became increasingly difficult and dubious and the set of data to process became voluminous [6].

However, the decision-makers should access data to analyze it and extract relevant information to take good decisions in short delays, to reply to the needs of the customers and to face the competitors.

Therefore, each business should transform data in useful knowledge in order to:

- Know the behaviours of its customers.
- Reply to the customer in a short time.
- Optimize the sales by customer.
- Identify the more profitable customers.
- Identify the potential customers and study the other markets.

This customer relationship management needs the use of tools such as statistical ones in order to determine the different segments of customers, the more adequate targets, the strongest needs, the privileged contact canal and moreover to foresee the escape risk of the customers towards the competitors [3]. The decision-makers in great distribution can invest in these tools [12]. They are able to have data warehouses. However, supermarkets are not big enough to invest in these tools. For this reason, we were interested by supermarkets and we think that clustering, which is an approach used in data mining, seems to be well adapted for the segmentation of supermarkets' customers. In what follows, we present an incremental clustering method which enables to cluster incrementally customers, by analyzing their properties and their purchase behaviours. It helps the decision-makers to understand their customers, to satisfy them by anticipating their needs and their expectations and to elaborate solid fidelity links with them.

2 Clustering

Clustering [7] consists in grouping together the elements having a similar behaviour in non predefined groups. The clusters can be mutually exclusive and exhaustive, and they can be defined by richer representations: hierarchical or overlapping.

Clustering is not a supervised approach and can be useful to discover a hidden structure allowing the improvement of the results of the other supervised methods (classification, estimation, prediction). It can be applied to all data types without doing a lot of transformations among them and the algorithms are easy to implement. However, the quality of the resulting clusters is very dependent of the similarity measure since the clusters must contain objects having a big degree of similarity (maximisation of the similarity intra-clusters) and they must be wide (minimisation of the similarity inter-clusters) [1, 5]. Therefore, this task is delicate especially when the data are of different types. In the other hand, a

clustering method, as k-means method [7], depends on the choice of parameters, in particular the choice of the number k of clusters, to produce good results and so to have good clustering. At last, it is difficult to interpret the results, thus, giving a meaning to the obtained clusters.

Incremental clustering enables creating clusters incrementally by opposition to non incremental clustering [11]. In fact, when using an incremental clustering method, one does not need to apply the method on all the initial data but only on added or deleted elements and the previous clusters.

Among the approaches of incremental clustering, we can refer to the algorithm of Godin [9], incremental DBSCAN [5], COBWEB algorithm [10], Classit system [8] and Adeclu system [4].

3 Incremental clustering for supermarkets

The proposed incremental clustering approach allows clustering supermarkets' customers according to their purchase behaviour and their properties. It allows replying to certain questions, as:

- Who are the customers of the supermarket?
- Which items do they look for?
- Which are the common characteristics between the customers who buy an item A_i with a given frequency?

In order to reply to these questions that could help the decision-makers to better build fidelity relation with their customers, we proceed according to two successive processes: a process of pre-treatment that transforms the brute data furnished by the supermarket, followed by a clustering process.

3.1 Pre-treatment process presentation

The pre-treatment process is used to prepare the brute data which contain the customers' properties and behaviours in order to cluster them such as it will be possible to understand better the customers' needs. This process includes two steps:

- The first one consists in summarizing every customer transactions while determining the total number of the transactions that he carried out as well as the purchase frequencies of the different bought items in comparison with the transactions that he carried out.
- The second step consists in proceeding by item. For every item Ai, chosen by the decision-maker, the customers that buy it with a frequency α superior to a fixed parameter are determined.

To illustrate this process of pre-treatment, we present an example of four customers, each one is described by four properties or attributes as follows:

Customer	P1	P2	P3	P4
0100	High	1	Adult	0
0012	High	1	Young	1
0053	Low	0	Old	0
1106	Medium	0	Adult	1

Where:

P1: Income which can be low, medium or high.

P2: Car owner which can take two values, 1 for yes, 0 else.

P3: Age slice which can be young, adult or old.

P4: Married which can take two values, 1 for yes, 0 else.

We suppose that the transactions carried out by these customers are as follows:

Transaction	Customer	A1	A2	A3	A4	A5	A6
1	0100	1	1	0	0	1	1
2	0012	1	1	1	1	1	1
3	0012	1	1	1	1	1	1
4	0012	1	1	1	1	1	1
5	0012	1	1	1	1	1	1
6	0053	1	0	0	1	1	1
7	0053	1	1	0	0	0	1
8	0053	1	1	0	0	0	1
9	0053	1	1	1	1	1	1
10	0053	1	0	0	1	1	1
11	1106	1	1	0	0	1	1
12	1106	1	1	0	0	1	1

Where:

- A1: Bakery item,
- A2: Butcher item,
- A3: Cheese item,
- A4: Frozen item,
- A5: Fish item,
- A6: Vegetable and fruit item.

First step: While applying the first step of the process of the pre-treatment on the transactions, we find the following data:

Customer	Number of Transactions	A1	A2	A3	A4	A5	A6
0100	1	1	1	0	0	1	1
0012	4	1	1	1	1	1	1
0053	5	1	0.6	0.2	0.6	0.6	1
1106	2	1	1	0	0	1	1

These data indicate, for every customer, the number of transactions that he carried out as well as the

purchase frequency of the items bought in comparison with all the transactions that he carried out. This frequency varies between 0 and 1. For example, the customer 0053 carried out 5 transactions and he bought 3 times on 5 the item A2; then his purchase frequency of A2 is equal to 0.6.

Second step: At this step, the decision-maker must choose an item Ai as well as a purchase frequency α for this item. The customers that have a purchase frequency for this item superior to α are then presented.
For example, the customers 0012 and 0053 verify for the item A3 a purchase frequency of 0.2.

3.2 Clustering process presentation

After releasing the customers that have a purchase frequency superior to α for a given item, we look for all the common characteristics to these customers in order to understand their purchase behaviours. Thus, we operate in two steps [2]:
- The first step consists in comparing the initial data and in releasing the similarities between the different customers.
- The second step consists in generating clusters of customers, based on their properties and the transactions that they carried out. The update of these clusters is incremental. Thus, the insertion, the delete and the modification of a customer and/or of a purchase transaction imply an update of the previous clusters in contrast with the non incremental methods [1, 7] that recreate, in these cases, all the clusters.

3.2.1 Comparison step

The customers and their properties are the inputs of this step. These data are represented by triplets of the form <customer identifier, attribute, value>. A comparative study of the properties of these customers is realized. Thus, the triplets of the different customers, having the same attribute, are compared. If two triplets present the same value of the attribute then a new triplet is generated. This triplet is notes <?, attribute, value>, where ? can represent each of the two customers. This triplet is added to a binary matrix L if L does not contain a triplet with the same attribute and the same value. Otherwise, the list of the customers of the triplet already existing in L is updated.
To illustrate this step, the customers that have a purchase frequency of the item A5 superior to 0.7 are taken. These are the customers 0100, 0012 and 1106. A binary matrix P of triplets presents the properties of the customers. The elements to 0 of the matrix are not represented for clarity reasons.

Triplet	Customer 0100	Customer 0012	Customer 1106
<0100,P1,High>	1		
<0100,P2,1>	1		
<0100,P3,Adult>	1		
<0100,P4,0>	1		
<0012,P1,High>		1	
<0012,P2,1>		1	
<0012,P3,Young>		1	
<0012,P4,1>		1	
<1106,P1,Medium>			1
<1106,P2,0>			1
<1106,P3,Adult>			1
<1106,P4,1>			1

The binary matrix L generated after the comparison is represented below.

Triplet	Customer 0100	Customer 0012	Customer 1106
< ?,P1,High>	1	1	
<?,P2,1>	1	1	
<?,P3,Adult>	1		1
<?,P4,1>		1	1

3.2.2 Clusters' extraction step

This step allows the extraction of clusters from the matrix L generated previously. Every cluster is in fact a rectangle.

Definition: Let R be a binary relation of E×F. A rectangle of R is the couple (A, B) such as $A \subseteq E$, $B \subseteq F$ and $A \times B \subseteq R$. A is the domain of the rectangle and B its co-domain.

In our case, each rectangle has a set of generated triplets as domain and a set of customers as co-domain and it allows defining a cluster of customers characterized by the same properties.
The clusters' extraction algorithm operates as follows:
- Every triplet of L is tested to see if it belongs to the domain of an already existing rectangle. In this case, it is deleted from the domain of this rectangle.
- If the customers linked to this triplet form the co-domain of an already existing rectangle then the triplet is added to the domain of this rectangle.
- If the customers linked to this triplet don't belong to any co-domain, then a new rectangle will be created. The triplet will form its domain and the customers that are linked to it its co-domain.

The clusters extracted from the matrix L released previously are defined below:

Cluster	Properties	Customer
G1	P1=High P2=1	0100 0012
G2	P3=Adult	0100 1106
G3	P4=1	0012 1106

Three clusters are extracted from the set of customers that buy the item A5 (fish items) with a purchase frequency superior to 0.7:
- From the first cluster, it is deduced that 66% of them have high income and own a car.
- From the second cluster, it is deduced that 66% of them are adults.
- From the third cluster, it is deduced that 66% of them are married.

For example, we can deduce from these clusters that fish items have high prices and then can't be bought by every body. If decision-makers want to attack a larger population with these items, they should present larger panoply with different prices. Other deductions can be also taken.

3.3 Presentation of the principal algorithms

Here are presented the two principal algorithms: the first one generates the matrix L and the second one generates the clusters.
Notations:
- O: the set of customers to compare.
- <key, attribute, value>: a triplet. There are two types of triplet: the initial ones and the generated ones. An initial triplet presents a relation between an attribute and a customer. A generated triplet presents a relation between an attribute and some customers who have the same value for this attribute.
- P: a binary bi-dimensional matrix. The lines are the initial triplets and the columns are customers.
- L: a generated binary bi-dimensional matrix. The lines are the generated triplets and the columns are the customers.
- O_j: the customer j of P or L.
- t_i: the triplet i of P or L.
- O_{ti}: the set of customers verifying ti of L.
- RE_k: the k^{th} rectangle
- Clusters: the set of all extracted rectangles.

3.3.1 Generation of the matrix L

This algorithm builds a binary matrix L while using the initial data presented in the binary matrix P. It is a question, first of all, of comparing the triplets of the matrix P which presents different customers and their properties. If two triplets present the same value for an attribute then a new generated triplet is created. This triplet is added in the relation L if this one does not contain a triplet with the same attribute and the same value for this attribute. In the other case, the list of the customers of the already existing triplet in L is updated. We notice that:
- The symbol "?" is used to represent the key of a generated triplet
- The function basic(t_i) returns true if t_i is an initial triplet, otherwise it returns false.
- The function selected(t_i) returns true if t_i belongs to an existing cluster, otherwise it returns false.

Algorithm 1: Generation of the matrix L
Input: P
Output: L
Begin
For all (t_i, O_j) ∈P **Do**
$P(t_i, O_j) \leftarrow 1$
$Basic(t_i) \leftarrow$ true
For all $O_k \in O$ such as $O_k \neq O_j$ **Do**
Search the triplet T of O_k such as $T_{attribute}=t_{i\ attribute}$ and basic(T)=true
If found **Then**
If $T_{value}=T_{i\ value}$ **Then**
t_n= <?, $T_{attribute}$, T_{value}>
If $t_n \notin L$ **Then**
Add t_n to L
$Basic(t_n) \leftarrow$ false
$Selected(t_n) \leftarrow$ false
End if
$L(tn, Ok) \leftarrow 1$
$L(tn, Oj) \leftarrow 1$
End if
End if
End

If m is the number of triplets of the matrix P and n is the number of customers of the matrix P then the complexity of this algorithm is $O(m^2 \times n^2)$.

3.3.2 Generation of clusters

This algorithm creates the set of clusters from the matrix L generated by the previous algorithm. It functions as follows:
- Each triplet of L is tested to see if it belongs to the domain of a rectangle (a cluster) already built. In this case, it is removed from the domain.

- We seek, then, if the customers related to this triplet belongs to the customers' set of a rectangle already built (thus to its co-domain) then the triplet is added to the domain of the rectangle, else a rectangle will be created with the triplet as the domain and the customers related to it as the co-domain.

It is noted that the rectangle (or cluster) generated by this algorithm is particular, because its domain is defined by a set of triplets which describe exactly the same customers. The purpose is to define the customers which present exactly the same properties and the same purchase behaviors.

Algorithm 2: Generation of clusters
Input: L
Output: Clusters
Begin
Clusters ← ∅
For all $(t_i, O_j) \in L$ **Do**
If Selected(t_i)=false **Then**
Selected(t_i) ← true
Search a rectangle RE_k in Clusters such as $t_i \in dom(RE_k)$
If exists **Then**
$Dom(RE_k) \leftarrow dom(RE_k)-\{t_i\}$
If $dom(RE_k)= \emptyset$ **Then** delete RE_k from Clusters
Search a rectangle RE_k in Clusters such as $cod(RE_k)=\{O_j / O_j \in O_{ti}\}$
If exists **Then**
$dom(RE_k) \leftarrow dom(RE_k) \cup t_i$
Else Create RE_k such as
$dom(RE_k)=t_i$ and
$cod(RE_k)=\{O_j / O_j \in O_{ti}\}$
Add RE_k to Clusters
End

If k is the number of generated triplets of the matrix L and n the number of customers of L then the complexity of the algorithm is $O(k^2 \times n)$.

3.4 Conclusions

An approach of incremental clustering was presented in this section. It allows clustering customers having the same properties and the same purchase behaviors. It determines the more bought items and the common characteristics between the customers that buy a given item with a purchase frequency given.
The principal advantages of this approach are the following ones:
- The use of incremental clustering algorithm.
- The simplicity.
- The automatic affectation to the clusters.
- The number of generated clusters is not fixed in advance.
- The definition of a distance measure is not necessary.
- The generated clusters can overlap.

The principal inconveniences of this approach are the following ones:
- Before clustering, data should be structured in a special form (as triplets).
- The clustering process is rigid since it clusters the customers having exactly the same common points; this can generate a loss of information that could be useful for the decision-makers.

4 Experimental study

To do an experimental evaluation of the approach presented above, we were confronted to two problems: a large number of Tunisian supermarkets don't have complete data on their customers and the supermarkets that have data on their customers consider these as confidential. Therefore, we decided to create a commercial base that simulates customer's properties and their purchases. It includes 100 customers, 52 attributes and 1000 transactions. In what follows, we study the impact of the variation of the customer's number on the evolution of the size of the matrix L (figure 1), on the creation time of the matrix L (figure 2) and on the creation time of the clusters (figure 3). All the tests were carried out on a Pentium IV 1.80 GHz computer, with 256 KB memory.
We notice that the experiments are conforming to the theoretical study.

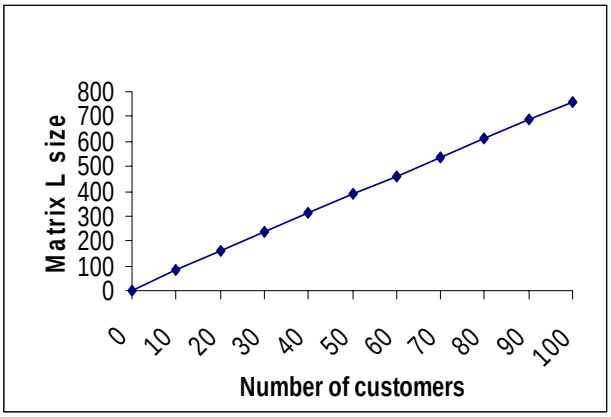

Fig.1 Evolution of the size of the matrix L according to the number of customers

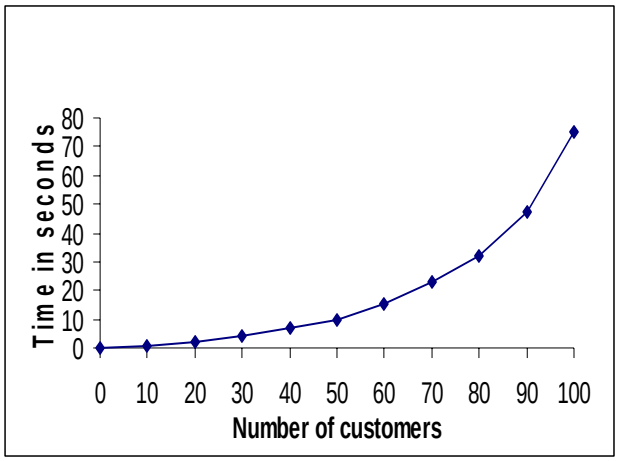

Fig.2 Evolution of the creation time of the matrix L according to the number of customers

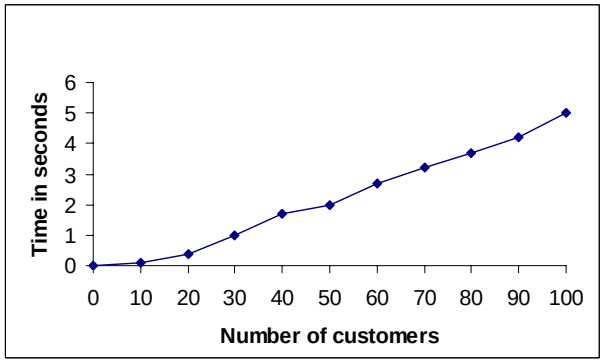

Fig.3 Evolution of the creation time of the clusters according to the number of customers

5 Comparative study

In this section, we compare the results obtained using the approach presented above and the algorithm of Godin applied to the same experimental base presented in section 4.
We notice that:
- The number of clusters generated by the approach presented above is lower than the number of clusters generated by the algorithm of Godin.
- The execution time of the approach presented is lower than that of Godin.

Figure 4 compares the variation of the execution time for both, the algorithm of Godin and the presented approach. We notice that the variation in execution time increases proportionally at the number of customers. Moreover, it is noted that the execution time of the algorithm of Godin is a little more important than the execution time of the presented approach. This can be explained by the difference of the number of produced clusters.

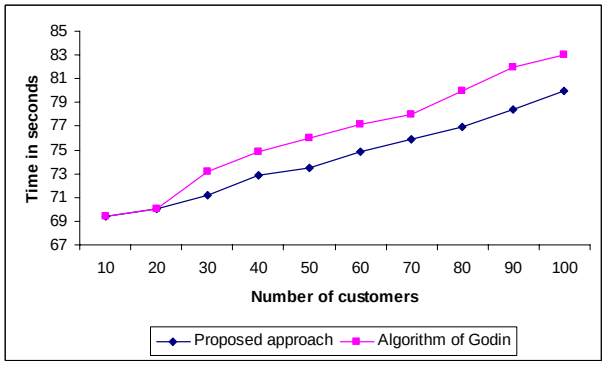

Fig.4 Evolution of the execution time according to the number of customers

Figure 5 illustrates the evolution of the number of clusters generated according to the number of customers. We notice that the number of clusters generated by the algorithm of Godin is very important compared to the number of the clusters generated by the presented approach.

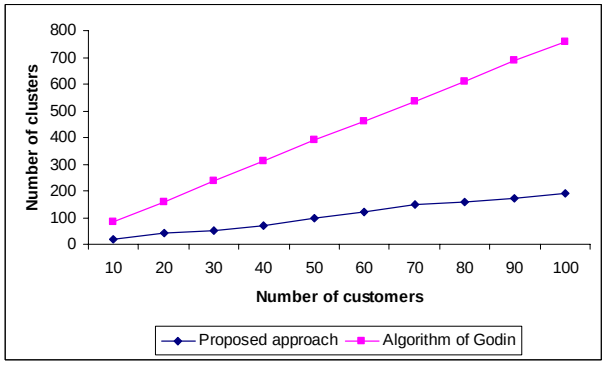

Fig.5 Evolution of the number of clusters according to the number of customers

We remark that even if the number of clusters is more important with the algorithm of Godin, thus extracted information is more important, however time necessary is more important than in the presented approach and the analysis of these clusters takes more time. This question can be put: can decision-makers spread a lot of time to get more knowledge, or they can be satisfied with less relevant knowledge but got rapidly?

6 Conclusion

In this paper, we presented an approach of incremental clustering related to commercial transactions, made of the properties of the customers as well as items bought by them. It operates in two steps: a pre-treatment step which enables transforming the raw data provided by the supermarkets before clustering and a clustering step which creates a new type of rectangle, each one is a cluster of customers.

The knowledge extracted can be of help to the decision-makers to understand the behaviour of the customers and thus to take good decisions.

We carried out an experimental study on a base of test created and we compared the results with those obtained using the algorithm of Godin [9]. We deduced that the execution time and the number of generated clusters are lower using the presented approach then using the algorithm of Godin.

For a better evaluation of the presented approach, we envisage to test it on a real basis. We also envisage modifying the approach to have a more flexible clustering process using a certain degree of similarity between the customers of the same cluster.

References:

[1] J. P. Auray, G. Duru et A. D. Zighed, *Analyse de données multidimensionnelles: Les méthodes de structuration*, Editions Alexandre Lacassagne, Lyon, 1990.

[2] H. Bélaïd Ajroud, *Abstraction rectangulaire des systèmes, des programmes et des connaissances*, Thèse de doctorat, Université des Sciences des techniques et de Médecine de Tunis, Juin 1999.

[3] A. Berson, S. Smith, and K. Thearling, *Building Data Mining Applications for CRM*, McGraw Hill, 2000.

[4] C. Decaestecker, *Apprentissage en Classification Conceptuelle Incrémentale*, Thèse de Doctorat, Université Libre de Bruxelles, Belgique, 1991.

[5] M. Ester, H. Kriegel, J. Sander, M. Wimmer, and X. Xu, *Incremental Clustering for Mining in a Data Warehousing Environment*, Proc. 24th Int. Conf. Very Large Data Bases, 1998.

[6] http://www.assurisk.com/exposes/gestion/crm.htm, *Présentation du CRM (Customer Relationship Management)*.

[7] http://www.grappa.univ-lille3.fr/polys/fouille/sortie005.html, *Tâches et Techniques du Data Mining*.

[8] J.H. Gennari, *An experimental study of concept formation*, Doctoral Dissertation, Department of Information and Computer Science, University of California, 1990.

[9] R. Godin, R. Missaoui, and H. Alaoui, Incremental formation algorithms based on Galois (concept) lattices, *Computational Intelligence*, N°.11(2), 1995, pp. 246_267.

[10] P. Langley, D. Fisher, and J.H. Gennari, *Models of incremental concept formation*, J. Carbonelle, ed., Cambridge, Mass.: MIT Press, 1990.

[11] H. Mannila, Global and local methods in data mining: basic techniques and open problems, *ICALP 2002, 29th International Colloquium on Automata, Languages, and Programming*, pp. 57-68, Malaga, Spain, July 2002.

[12] Wal-Mart: supertechnologie et vieilles recettes, *Management*, N°65, pp42-43, Juillet 2000.