Third International Conference
Modelling and Development of Intelligent Systems
October 10 - 12, 2013
“Lucian Blaga” University
Sibiu - Romania

Comparative Study in Building of Associations Rules from Commercial Transactions through Data Mining Techniques

Mircea-Adrian MUŞAN, Ionela MANIU

Abstract

In this paper we have built processes for extracting data mining association rules based on frequent sets of articles from commercial transactions. We used as a working data set a database created from online retail transactions. Based on the particularities of processes built, we performed a statistical analysis to illustrate the efficiency, precision and accuracy of data mining techniques used.

1 Introduction

The growing interest for Data Mining domain can be motivated through the pressing need, common to many areas of reference, to describe, to model and, especially, to understand large sets of data.

The process of knowledge discovery is equally old as the cerebral man. Since the discovery of fire and reaching out to the current studies on marketing, man made "data mining" without realizing it. Today, aided by tremendous computing power of computers, it can now adventure in exploring information by using the most effective means of working with existing data.

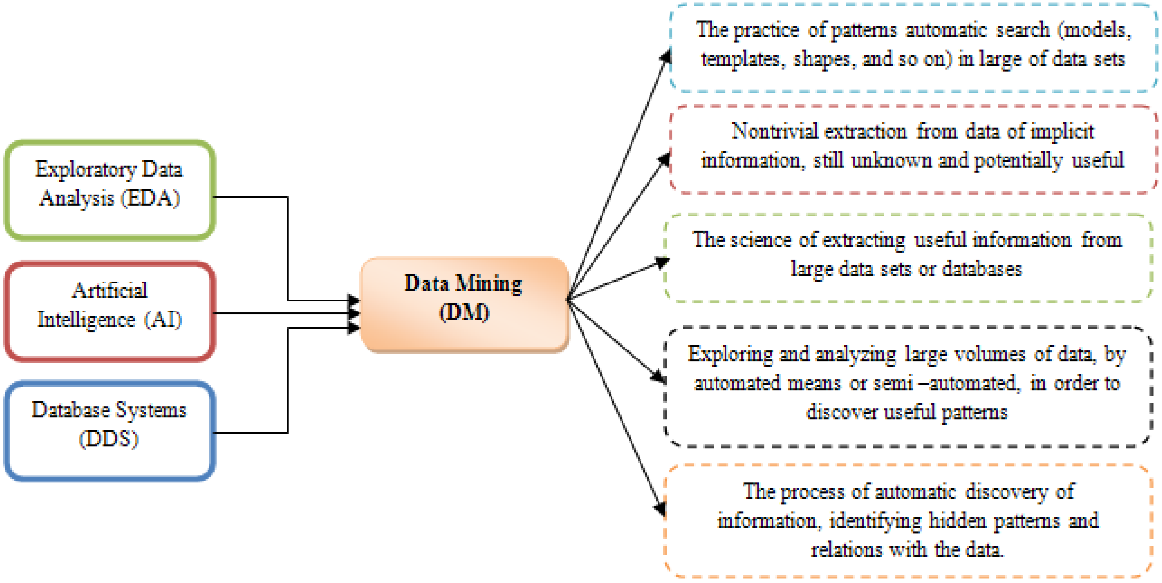
Exploratory Data Analysis (EDA)
Artificial Intelligence (AI)
Database Systems (DDS)
Data Mining (DM)
The practice of patterns automatic search (models, templates, shapes, and so on) in large of data sets
Nontrivial extraction from data of implicit information, still unknown and potentially useful
The science of extracting useful information from large data sets or databases
Exploring and analyzing large volumes of data, by automated means or semi –automated, in order to discover useful patterns
The process of automatic discovery of information, identifying hidden patterns and relations with the data.

Figure 1 – Roots and significance of data mining

As can be seen in *Figure 1*, it is difficult to formulate a single definition for data mining. On base of term roots (presented in *Figure 1*), namely, exploring data analysis, artificial intelligence and database systems, the most frequently encountered significance for the concept of "data mining" is, in a few words, "knowledge-discovery in databases" (KDD), as it is named in work [1]. Another definition, in the same work, is “extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data”.

A significant category of data mining techniques is that of mining frequent patterns, associations and correlations. Algorithms built for association rules are very useful from the perspective of marketing, because they develop methods for finding customers shopping patterns [6]. Applications of these special techniques are in basket data analysis, cross-marketing, catalogue design, sale campaign analysis, click stream of web logs analysis, and DNA sequence analysis [1]. From marketing perspective, the workings of these techniques are simple: purpose is to find correlations between articles sold. Association rules are based on two measures which quantify the support and confidence of the rule for a given data set.

The use of these techniques is to find trends and correlations in databases, which helps experts to take correctly and efficiently decisions in the future.

The second section of this paper presents an experiment based on the algorithm used; the third describes the processes constructed and the results obtained. A statistical analysis of the results was performed in the last section.

2 Experiment based on FP-Tree algorithm for our data set

The database after which we will make the processing is described in *Section 3* of this paper. To extract association rules from the products traded we chose operation by FP-Growth technique. The FP-Growth algorithm, that means Frequent Pattern Growth Algorithm, was developed by J. Han, H. Pei, and Y. Yin [8]. This efficient, fast and scalable algorithm [6] is a method for mining the complete set of frequent patterns by pattern fragment growth, using an extended prefix-tree structure for storing essential information about frequent patterns, named FP-tree [8].

Among the advantages offered by this algorithm, we can mention that it is one of the fastest in obtaining association rules, as J. Han wrote in his paper [8], that the FP-Growth algorithm has better performance than AprioriAlgorithm [3], Tree-Projection [9], RElim [10] or Eclat [11]. It also has disadvantages, namely, it is more difficult to implement that other approaches like a complex data structure and an FP-tree, it can need more memory than a list of transactions [2].

In our case study we randomly selected a set of ten transactions made. In order to not have a very wide range of products, articles of transactions we considered as being the name of category from which these products belong. Thus, if in a certain chosen transaction we will find at least two products from the same category, we keep only one item, namely, the name of that category. After all this, we obtained the following table:

Tr. ID	**List of product categories**
0	Hair, Women fragrance, Tools and brushes
1	Skin care, Women fragrance
2	Skin care, Men fragrance
3	Hair, Men fragrance, Women fragrance, Bath and body
4	Women fragrance, Bath and body
5	Skin care, Men fragrance
6	Hair, Skin care, Women fragrance
7	Men fragrance, Women fragrance
8	Skin care, Men fragrance, Bath and body
9	Hair, Skin care, Women fragrance

Table 1 – Set of transactions chosen by category name

After establishment of the list of transactions, it moves to the next level, namely, determination of frequency of individual items. In our case we have the next list: *Hair* – 4, *Skin care* – 6, *Men fragrance* – 5, *Women fragrance* – 7, *Bath and body* – 3 and *Tools and brushes* – 1.

The next step is that of sorting descending items in transactions and removing those items that are infrequent for the parameters chosen in our case, in our case *Tools and brushes*. After that, the transactions are sorted lexicographically in ascending order. This is the last step before construction of frequent patterns tree. The result is presented in *Figure 2*.

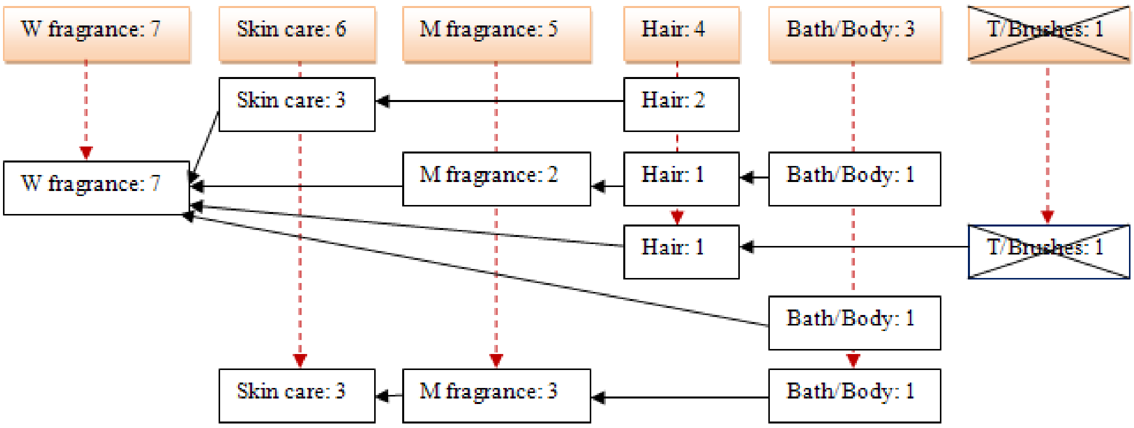

Figure 2 – FP Tree. Representation of transactions

The results obtained using FP-Tree algorithm, for our experiment, are presented in *Table 2*. For this, a value for minimum support (50%) is selected, relevant for our case. For establishing frequent sets of items we have chosen the same display mode of results like application RapidMiner.

Size	Support	Item 1	Item 2	Item 3
1	0.700	Women fragrance		
1	0.600	Skin care		
1	0.500	Men fragrance		
1	0.400	Hair		
1	0.300	Bath and body		
2	0.300	Women fragrance	Skin care	
2	0.200	Women fragrance	Men fragrance	
2	0.400	Women fragrance	Hair	
2	0.200	Women fragrance	Bath and body	
2	0.300	Skin care	Men fragrance	
2	0.200	Skin care	Hair	
2	0.200	Men fragrance	Bath and body	
3	0.200	Women fragrance	Skin care	Hair

Table 2 – Displaying of frequent sets

3 Presentation of the process built

3.1 Description of the process

In our process programming we used RapidMiner. RapidMiner assures data mining and machine learning procedures, such as: pre-processing and visualization of data, transformation, modelling, evaluation, and deployment of data. RapidMiner is written in the Java programming language and uses learning schemes and attribute evaluators from the Weka machine learning environment and statistical modelling schemes from R-Project [7].

RapidMiner contains a collection of modular operators which allow the design of complex processing for a large number of data mining problems. The most important characteristic of

RapidMiner is the ability to imbricate operators' chains and building trees of complex operators. To support this feature, data core of RapidMiner acts as a databases management system.

In order to program the desired process for analyzing associations of appearances frequent transaction sets, we used a dataset of an e-commerce company, operating in the field of perfumery and personal care products.

We chose a database of a company that operate exclusively online for several reasons, namely: the first reason is that the data set proposed for marketing is homogeneous and the second is that most transactions are composed of at least 2 – 3 products, and that from economic reasons related to saving the transport costs, promotions on the number of products, reasons helpful in analyzing associations based transactions.

The dataset used (file named Fragrance.xls) has over 1,500 lines, overall 500 transactions, and the following structure of fields:

- *Current number of record* (auto number)
- *ID of product* (a numerical value between 101 and 999)
- *Name of product* (nominal value)
- *ID of category* (a numerical value between 1 and 99)
- *Name of product's category* (nominal value)
- *ID of transaction* (a numerical value)

Based on the dataset described above, we developed a data mining process developed using Rapid Miner, which will determine sets of frequent appearances from transactions, on which are generated association rules. Process built is based on the drawings of other processes constructed through the work [3] [4], being shown in *Figure 3*.

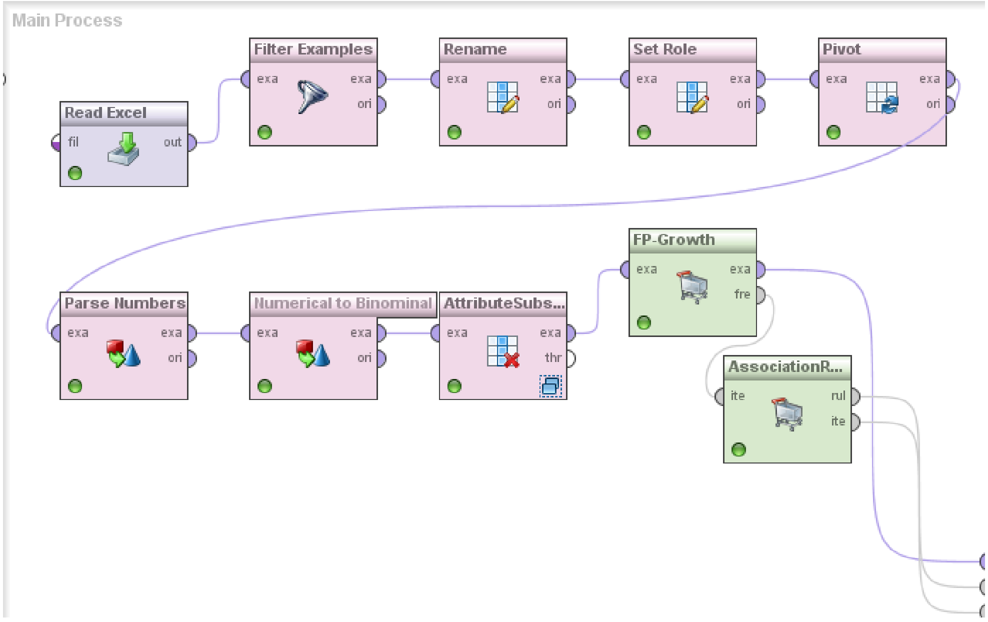

Figure 3 – The RapidMiner process for determining sets of frequent appearances and association rules generated

For this process writing, created by facility GUI of RapidMiner and refined programmatically through adequate XML code, for reasons of space, it will not be exposed in this article, we used the following operators:

- **Read Excel**, through which we took the dataset, but eliminating those elements deductible, namely the information about product categories, and that is not to force later shift of all attributes in a binomial form, which is essential in process development .
- With help of **Filter Examples** we have split the initial data set in several samples, depending on the selected product categories and units of time in which transactions were made. Thus, many results were generated, which are interpreted and analyzed in the following sections.

- **Rename** is used to rename fields, which by their nature, participate directly in the results, and as such, would hold their visibility.
- **Set Role** is used by RapidMiner to change the role of one or more attributes. In our case we put value *ID of transaction* to field ***attribute name***, ***target role*** received value *id*. Through option ***set additional roles*** we have established *Name of product* as being *regular* type.
- **Pivot** is an important operator of this process and we used it to rotate the example set by grouping multiple examples of same groups to single examples. By option ***group attribute*** we selected the field *ID of transaction*, by ***index attribute*** we have chosen the field *Name of product* and through ***weight aggregation*** we selected the option *count*.
- **Parse Number** is an operator auxiliary in this process, which we wrote for applying the next operator from our process for all data.
- **Numerical to Binomial** changes the type of the selected numeric attributes to a binominal type. It is an essential operator from this process, because the operator with name ***FP-Growth*** works only with binomial values. Because we applied before the operator ***Parse Number***, in this case we chose the option *all* for the option ***attribute filter type***.
- **Attribute Subset**, through which a subset is selected, composed of one or more attributes, from the input dataset and applies the operators in its subprocess on the selected subset.
- **FP-Growth** is a central operator of our construction. It calculates all frequent itemsets from the given dataset using the *FP-tree* data structure. The range of values within which we chose *minimum support* for establishing frequent sets of items is described in *Section 4*.
- **Association Rule Generator** (**Create Association Rule**) was written to obtain the association rules generated based on frequent occurrences of articles in transactions as they have been previous outcomes by using of operator, FP Growth. Data related to the values received by *minim confidence* attribute will be reported in *Section 4*, these constituting the support for the hypothesis of statistical analysis based on the results obtained. For this operator we have selected the output port *ite*, in order to see the frequent item sets obtained by FP-Growth operator.

3.2 The results obtained

In order to show the results, we selected a value for minimum support and one for minimum confidence. For reasons of space, we chose the maximum values from the range presented in *Section 4*, resulting the case with the fewest rules of associations obtained. So, we have chosen minimum support as 12%, and minimum confidence as 50%, as an example. The results generated by RapidMiner can be observed in the following list. It is quite obvious that if we choose smaller values for the parameters indicated, the number of generated rules will be greater, but including among themselves the rules presented below.

```
Association Rules
[Body lotions & body oils] → [Men's perfume] (confidence: 0.500)
[Shampoo, Conditioner] → [Gift sets] (confidence: 0.500)
[Shampoo] → [Conditioner] (confidence: 0.517)
[Teeth whitening, Hair color] → [Cellulite & stratch marks] (confidence:
0.524)
[Hand & foot cream] → [Men's perfume] (confidence: 0.560)
[Eye brushes] → [Women's perfume] (confidence: 0.562)
[Men's perfume, Teeth whitening] → [Night cream] (confidence: 0.571)
[Lip brushes] → [Women's perfume] (confidence: 0.605)
[Eye brushes] → [Face brushes] (confidence: 0.625)
[Conditioner] → [Shampoo] (confidence: 0.667)
[Teeth whitening, Moisturizer] → [Hair color] (confidence: 0.688)
[Men's perfume, Night cream] → [Teeth whitening] (confidence: 0.750)
[Day cream, Hair color] → [Teeth whitening] (confidence: 0.786)
[Gift sets, Shampoo] → [Conditioner] (confidence: 0.789)
[Face brushes] → [Eye brushes] (confidence: 0.833)
[Gift sets, Conditioner] → [Shampoo] (confidence: 0.938)
```

List 1 – The results obtained by RapidMiner on our data set

These results were obtained from *Text View* option, generated by RapidMiner. If it is desired a visualisation more complete of the results, one can choose *Table View* option. For example, first association rule has *Support* 0.050, *LaPlace* 0.955 and *Gain* -0.150.

At the results interpretation, the following rules can be obtained: "for the premise *Body lotions & body oils* at least 12% of the customers of this product always buy *Men's perfume*, obtained by the attribute conclusion" or "at least 12% of customers of *Shampoo* and *Conditioner* always buy *Gift sets*".

4 Statistical representations and analysis based on the results

In this paper, for the experimental part, we used Rapid Miner pre-programmed process presented in *Section 3*, executing it several times. Execution numbers has been determined by a variety of values for the process key attributes: set minimum support and guaranteed minimum confidence. For our experiment, we used a minimum support of a range of values from the set [0, 0.12], which are in arithmetic progression with the ratio 0.02, each of which is assigned to a value from the set [0, 0.5] in arithmetic progression with ratio 0.05, for minimum confidence attribute. Thus 78 runs resulted in the process, enough to carry out a statistical experiment, based on our inputs.

Then we divided the data set in three equal parts according to chronological data entry, relying on a statistical survey of consumer behaviour on basis of joint sets transactions frequent occurrences in certain units of time.

The process was repeated for each of the three cases, yielding filtering through *Filter Examples* operator, so that attribute *condition class* receive value *attribute_value_filter* and *parameter string* corresponding filter condition. Running three new executions has been done using the same value range *minim Support – minim Confidence* as in process running on the full data set.

Based on ideas presented in the article [5], we used as a premise in our work, in statistical evaluation, the maximum of items present in the transaction and the average number of items per transactions. This gave the following table:

Transaction	Trial run on entire data set	Trial running on first data set	Trial running on second data set	Trial running on third data set
Maxim	9	6	6	9
Mean	3,340796	3,306931	3,207961	3,507538

Table 3 – The evidence of items from transaction

For the entire database a negative correlation was obtained linking the parameters support (r = -0.262, 0.043), confidence (r = -0.902, p = 0.000) and number of association rule, with a stronger link between confidence and rules number.

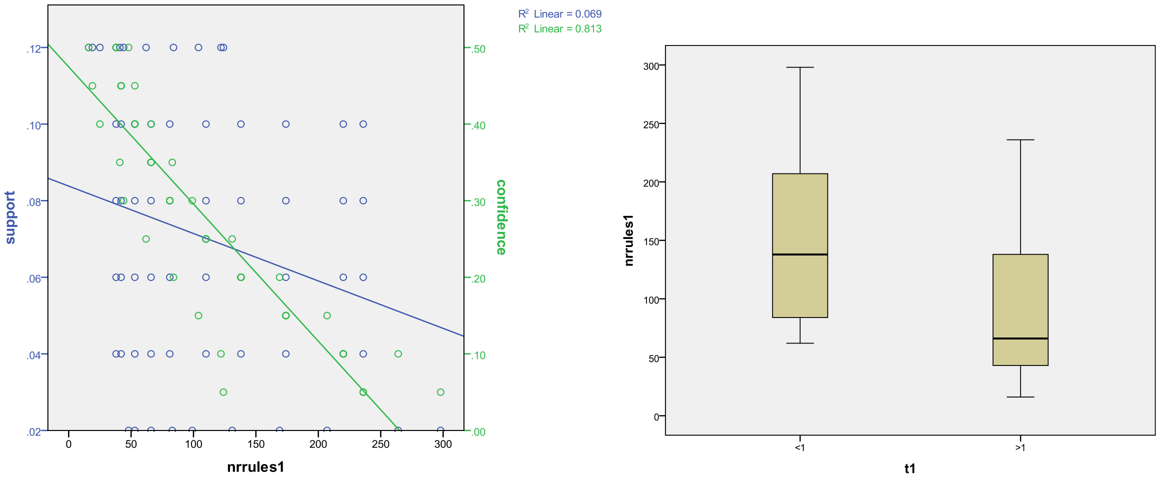

Figure 4 – Number of association rules vs. support – confidence and time, for the entire database

In case of database division, when the number of association rules is smaller, a negative correlation was kept between both parameters support (r1 = -0.505, p = 0.000), the confidence (r2 = -0.717, p = 0.000) and the number of association rules, having strong correlation between the confidence and the rules number. For a higher number of association rules, negative correlation are preserved between both parameters (r1 = -0.586, p = 0.000, r2 = -0.323, p = 0.012) and the number of association rules, but the link is strongest between the support and the association rules number.

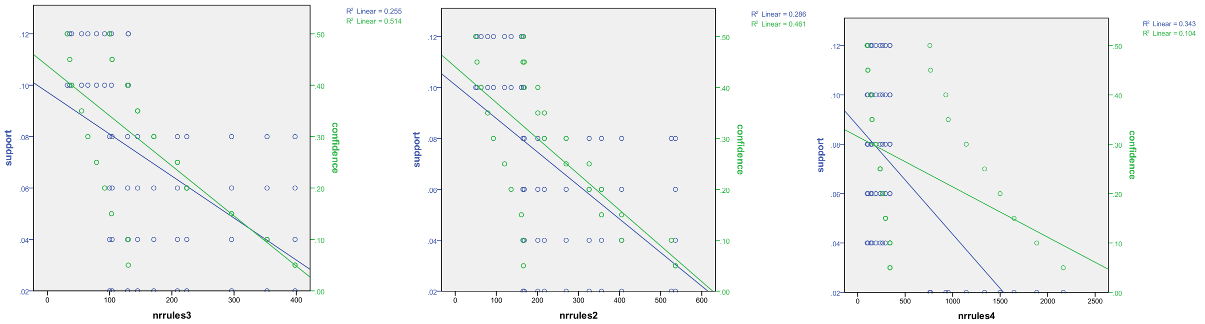

Figure 5 – Number of association rules vs. support – confidence, for database divisions

It is observed that for the same number of transactions (taken on time series) with the same products range, and the same *minim Support – minim Confidence*, a larger generated number of association rules does not involve a higher execution time, as can be seen in *Figure 6*, where on x-axis is represented the execution time and on y-axis the number of association rules generated.

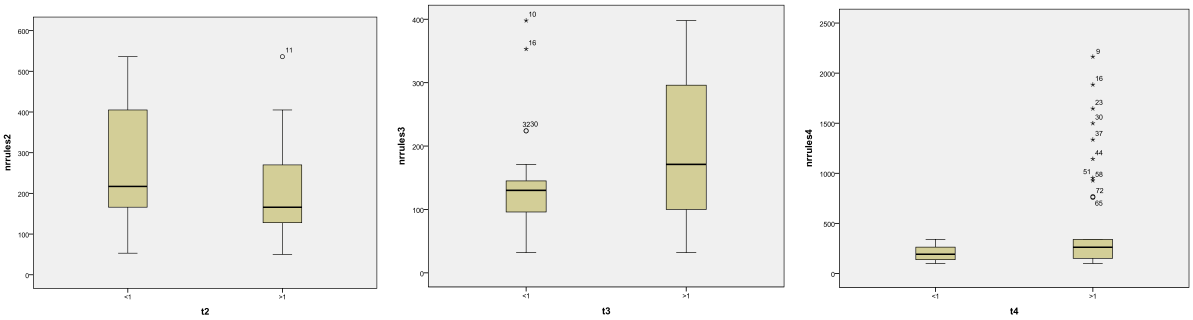

Figure 6 – Number of association rules vs. time, for database divisions

5 Conclusions

Techniques for determining the association rules are some very powerful tools in making decisions of marketing. Thus, based on them can be established some promotional packages, certain promotions, web page layout or arranging on the shelf, or simply, can track some trends of consumers. About the execution time of the association rules processes, the statistic analysis concludes that a larger generated number of rules does not involve a higher execution time.

References

[1] Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques*, 2^{nd} *ed.*, The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor Morgan Kaufmann Publishers, March 2006. ISBN 1-55860-901-6, http://www.cs.uiuc.edu/homes/hanj/bk2/slidesindex.htm

[2] Ch. Borgelt, *Frequent Pattern Mining*, Intelligent Data Analysis and Graphical Models Research Unit European Center for Soft Computing, 33600, Mieres, Spain, 2005

[3] Daniel Hunyadi, *Performance Comparison of Apriori and FP-Growth Algorithms in Generating Association Rules*, The 5th European Computing Conference (ECC' 11), Paris, France, April 28-30 2011, pp. 376-381, 978-960-474-297-4

[4] Mircea Muşan, *Versatile integration of data mining techniques of description and prediction in Web informatics systems of Business Intelligence*, Second International Conference "Modeling and Development of Intelligent Systems", Sibiu, Romania, September 29 – Octomber 02, 2011, pp. 97-104, ISSN 2067-3965, ISBN 978-606-12-0243-0

[5] Pratiksha Shendge, Tina Gupta, *Comparative Study od Apriori & FP Growth Algotithms*, Indian Jounal of Research, PARIPEX, Volume: 2, Issue: 3, March 2013, pp. 20-22, ISSN 2250-1991

[6] Jerzy Korczak, Piotr Skrzypczak, *FP-Growth in Discovery of Customer Patterns*, Data-Driven Process Discovery and Analysis Lecture Notes in Business Information Processing Volume 116, 2012, pp 120-133, Print ISBN 978-3-642-34043-7, Online ISBN 978-3-642-34044-4

[7] "*RapidMiner*", Rapid-i. Retrieved 7 March 2011, http://rapidminer.com/products/rapidminer-studio/

[8] J. Han, H. Pei, and Y. Yin, *Mining Frequent Patterns without Candidate Generation*, In: Proc. Conf. on the Management of Data (SIGMOD'00, Dallas, TX). ACM Press, New York, NY, USA 2000

[9] Agarwal,R.C., Agarwal C.C., Prasad, V.V., *A Tree Projection Algorithm For Generation of Frequent Itemsets*. Journal on Parallel and Distributed Computing, vol. 61, 2000

[10] Christian Borgelt, *Keeping Things Simple: Finding Frequent Item Sets by Recursive, Elimination*, Workshop Open Source Data Mining Software (OSDM'05, Chicago, IL), 66-70, ACM Press, New York, USA 2005

[11] Zaki, M.J., Parthasarathy, S., Ogihara, M., Li, W., *New Algorithms for Fast Discovery of Association Rules*. ACM SIGKDD (1997)

Mircea-Adrian MUŞAN
"Lucian Blaga" University of Sibiu
Mathematics and Informatics
Sibiu, Street Ion Raţiu, No. 5
ROMANIA
E-mail: musanmircea@yahoo.com

Ionela MANIU
"Lucian Blaga" University of Sibiu
Mathematics and Informatics
Sibiu, Street Ion Raţiu, No. 5
ROMANIA
E-mail: ionela.maniu@yahoo.ro