Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Improving of e-business activities by building web applications with integrated data mining services

Mircea-Adrian Mușan, Iuliana-Maria Cândea

Abstract

Our work presents a way of integrating data mining techniques, written through Rapid Miner processes, into Web applications, from a marketing perspective for business efficiency. Through the application presented in this paper we outline the advantages of integration of data mining techniques into e-commerce systems, as a new element in the progress of using informatics into e-business activities.

1 Introduction

Data mining techniques used in computer applications for e-business activities opened new possibilities for handling information in real time. Informatics systems based on these techniques assist successfully entrepreneurs in making decisions to achieve a higher degree of economic efficiency at their organisation.

Data mining techniques offers a broad and useful perspective in developing and using information systems in the field of e-business. Diverse areas and purposes of use, together with the need to remote access, are elements of departure in this work.

Data mining is the process of extracting patterns from large data sets, by combining methods from statistics and artificial intelligence with database management [1][2]. With recent tremendous technical advances in processing power, storage capacity, and inter-connectivity of computer technology, data mining is seen as an increasingly important tool by modern business, to transform unprecedented quantities of digital data into business intelligence giving an informational advantage. It is currently used in a wide range of profiling practices, such as marketing, surveillance, fraud detection, and scientific discovery. The growing consensus that data mining can bring real value has led to an explosion in demand for novel data mining technologies [3].

It is difficult to formulate one single definition for data mining. In the Figure 1 we tried to extract more equivalent definitions. Most common significations for Data Mining are "knowledge-discovery in databases" (KDD) or "extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data" as it is named in work [4].

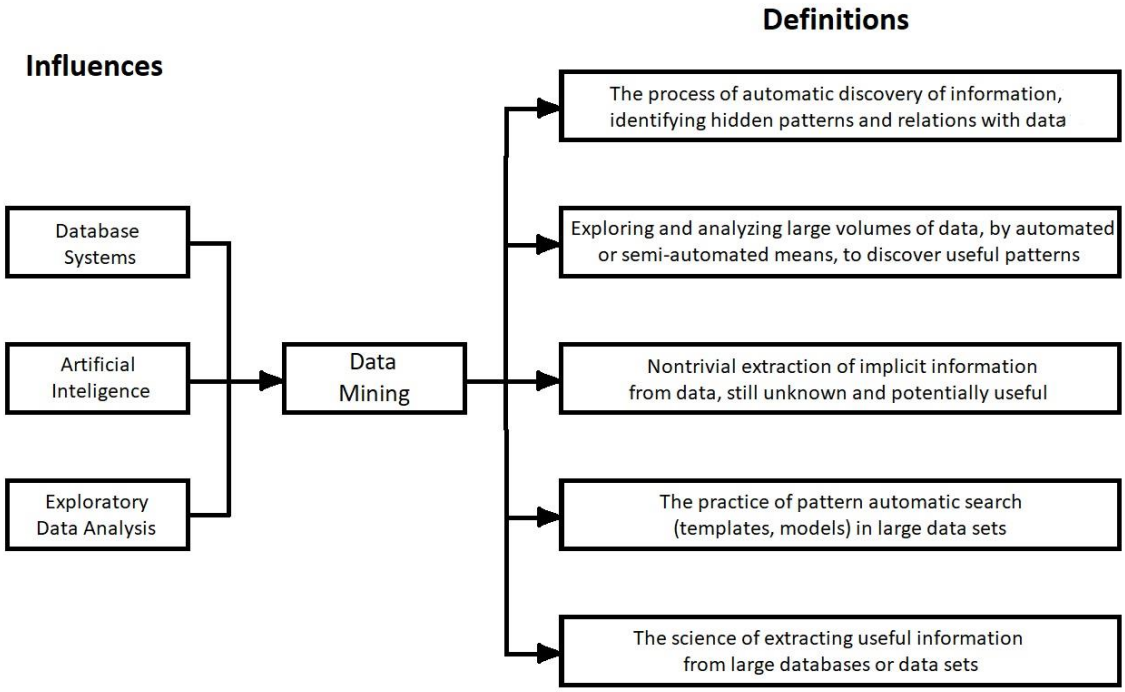

Figure 1 – Influences and definitions of data mining

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 [4]. In this paper we referred to that data mining component of association techniques and their analysis. We have created a remotely managed computer system, so that a Rapid Miner process, running through data mining operators and techniques, is written and uploaded to the application server and then accessed from our application, based on permanently updated data.

2 Process presentation

For determining the frequent item sets, we used the FP-Growth algorithm, which means Frequent Pattern Growth Algorithm, developed by J. Han, H. Pei, and Y. Yin [8]. This algorithm 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. This structure is known by the name of FP-tree. It is efficient, fast and scalable.
In the establishment of the association rules, we used the Apriori algorithm, written by Agrawal and Srikant in 1994 [9]. It determines the support of frequent sets of items by the method BFS (Breadth First Search). First, it determines the support of the sets one item, than with two items, continued recursively in the same way.
The RapidMiner environment contains a wide range of modular operators which allow the design of complex processing for a large number of data mining problems. An important characteristic of RapidMiner is the ability to imbricate operator chains and build trees of complex operators. We started from a process build through the work [5] but adapted it to a dataset stored in a MySQL database that can be stored on a server.

The dataset used in our process has the following structure of fields:

- *ID of movie* (a numerical value)
- *Name of movie* (nominal value)
- *ID of customer* (a numerical value)

Based on the dataset described above, we developed a data mining process using Rapid Miner, which will determine sets of frequent appearances from transactions, on which are generated association rules. Process built is shown in *Figure 2*.

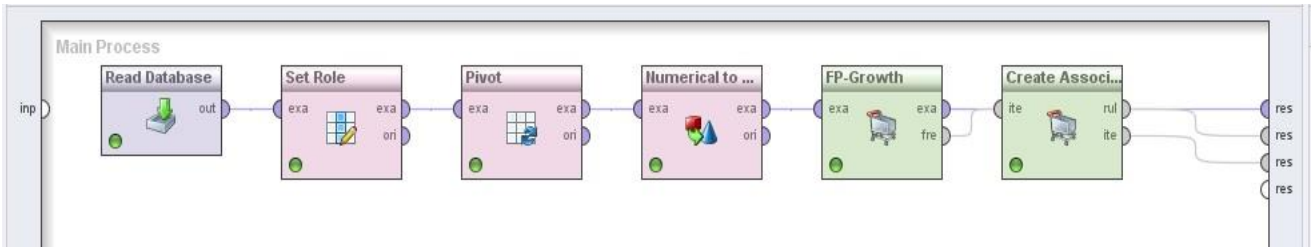

Figure 2 – The RapidMiner process for determining sets of frequent appearances and association rules generated with FP-Growth and Association Rules

For this process writing, created by facility GUI of Rapid Miner 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 Database** connects to a specified SQL database and reads an ExampleSet. In order to retrieve the data, a query can be specified. In our case, the query is:

```
SELECT `movies`.`movie_id` as MovieId, `transactions`.`customer_id` as CustomerId,
CONCAT(`movies`.` movie_id `, '_', `movies`.`name`) as MovieName
FROM `transactions`, `movies` WHERE `movie`.`movie_id` = `transactions`.`movie_id`
```

- **Set Role** is used by RapidMiner to change the role of one or more attributes. In our case we put value *ID of customer* to field ***attribute name***, ***target role*** received value *id*. Through option ***set additional roles*** we have established *Name of movie* 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 customer*, by ***index attribute*** we have chosen the field *Name of Movie* and through ***weight aggregation*** we selected the option *count*.
- **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. We chose the option *all* for the option ***attribute filter type***.
- **FP-Growth** is a central operator of our construction. It calculates all frequent item sets 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.
- **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 *minimum confidence* attribute, these constituting the support for the hypothesis of statistical analysis based on the results obtained.

3 Case study

In this part, we present how the Rapid Miner process mentioned in paragraph 2 can be applied into a real e-commerce application. Also, we explore how the library that realises the integration of Rapid Miner processes into Java applications can be used for a website created based on Java Server Pages technology.

3.1 The scenario

We have an online movie store. We need to make the most optimum associations, so the probability that the user that bought a movie would be interested in another suggested movie is as high as possible. The process presented in paragraph 2 in can be used to calculate which other movies are highly probable to be bought when one particular movie is bought, based on association rules, and also, what movies have the highest probability to be bought together, thus making possible to create and sell packages consisting of multiple movies.

3.2 The Application

The structure of the application is displayed in Figure 3.

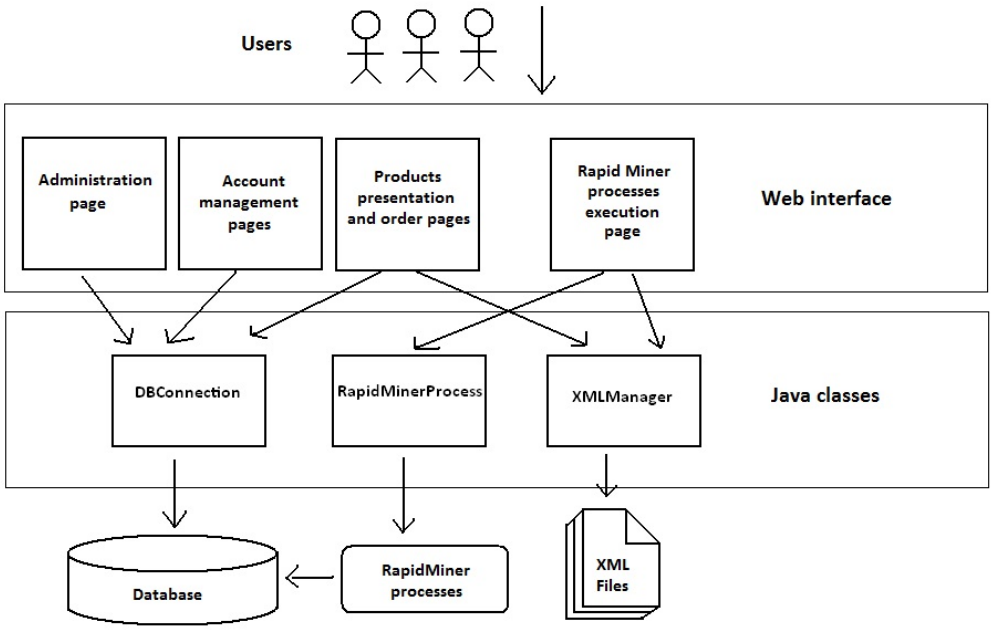

Figure 3 – Application structure

We want to point out the binding of the Rapid Miner process to the database that the application is build on, so every run of the process is using the most recent records. A particular Java class is used to change parameters and run the process, the methods being called by the users with administration rights on a special page.

3.3 Rapid Miner integration in Java class

Rapid Miner offers a package named com.rapidminer that can be included in Java projects. By the usage of this package, the user can import existing processes or breate new ones, change parameters, execute processes and display results. In this paragraph, we will present how we use in our application some important methods included in the package [7].

Initialize the Rapid Miner workspace:

```
RapidMiner.setExecutionMode(RapidMiner.ExecutionMode.COMMAND_LINE);
RapidMiner.init();
```

Import a process that is already created – Having the path of an existing process saved in a string variable named processRM, we create a new object of the Process class mapping the existing process by calling:

```
Process process= new Process(new File(processRM));
```

Retrieve an operator from a process – Having an object named process of type Process, to get an object of type Operator that represents the FP-Growth operator of the process, we call:

```
Operator operator = process.getOperator("FP-Growth");
```

Change a property of an operator – To set the minimum support of a FP-Growth operator, represented by an object named operator, to a value we have stored in a variable of type double named minSup, we execute the following line:

```
op.setParameter(FPGrowth.PARAMETER_MIN_SUPORT.Tools.formatNumber(minSup));
```

Run a process – The results of a process execution are stored in an object of IOContainer type. Having an object named process of class Process, the run method can be executed by calling:

```
IOContainer ioResult = process.run();
```

Parse process results – The result of a process execution, stored in an object of type IOContainer, are represented as a collection of objects implementing the IOObject interface. Each of the objects in the collection has a different type, depending on the operators present in the process. In our case, the second element will be an object of AssociationRules class, and the third element will be an object of FrequentItemSets class created by us, used to store the item name as string and the frequency as double.

- Parsing association rules:

```
IOObject result = ioResult.getElementAt(1);
List<String> associations = new ArrayList<String>();
if (result instanceof AssociationRules)
{
        String rules = result.toString();
        rules=rules.substring(rules.indexOf("\n")+1, rules.length());
        associations = Arrays.asList(rules.split("\r\n"));
}
```

- Parsing frequent item set:

```
IOObject result = ioResult.getElementAt(2);
List<FrequentItemSetClass> frequentItemSetDtos = new ArrayList<>();
if (result instanceof FrequentItemSets)
{
        for (FrequentItemSet fis : (FrequentItemSets) result)
        {
                FrequentItemSetClass frequentItemSetDto = new FrequentItemSetClass();
                frequentItemSetDto.setItem(fis.getItemsAsString());
                frequentItemSetDto.setFrequencyRatio(Tools.formatNumber((double) fis.getFrequency() /
                (double) ((FrequentItemSets) ioResult).getNumberOfTransactions()));
                frequentItemSetDtos.add(frequentItemSetDto);
        }
}
```

3.4 Interface for process execution

The section of the application used for executing Rapid Miner processes consists of two textboxes of numeric type, that can be used to change the minimum support and minimum confidence parameters, and three tables, that display the result of the process execution. In these tables we can see what movies are frequently bought together and what movies are most probably to be bought when one specific movie is bought. In Figure 4 we show the textboxes for the process inputs.

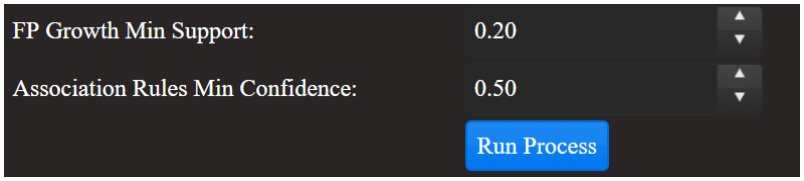

Figure 4 – Input fields for process parameters

The first input can be used to specify the mininum support parameter for the fp growth process, and the second numeric input can be used to specify the minimum confidence parameter for the assocation rules process.

Figure 5 – Process results and the way to display into application

In Figure 5 are presented together sections from the processes running page, movie information page that contains suggestions and packages, and home page which displays top movies of the moment.
In the left part of the figure is displayed a part of the detail page for the movie “Thor”. The middle part of the image contains two tables, representing the result of running the association rules process in the upper table and the result of running the fp growth process in the lower table. By looking in the upper table, we can see that the users who bought the movie “Thor” also bought „The Avengers and Iron Man, same things being suggested on the Thor detail page.
Also, by looking at the second table, we can see that the movie “Thor” was bought together with the movies “Iron Man”, “The Avengers” or both of them, so is more likely that other customers interested in the movie would buy a package containing these other movies.
In the right side of the figure, the upper part contains a section from the home page that displays the most popular movies since the last run of the processes. In the lower part, there is a table which display the movies that were bought most frequent, without being asociated with other movies, also coming as result from running the fp-growth process.

4 Conclusions

Data Mining Techniques for Associations Analysis is a modern and beneficial perspective in developing e-business activities, in particular the e-commerce component, through the marketing facilities offered. They enable both an activity of online marketing development through easy

access to products and the ability to make promotional packages as well as intelligent storage of products in warehouses. It is known that a smart, smart placement of products in a large deposit contributes to more efficient work, reducing time and spending on staff. In other words, the benefits are both on the customer side, through suggestions, promotional packs and time reduction, and on the vendor side, increasing employee productivity.
Integration of a Rapid Miner process into a Java application allows the user to change parameters as he wants. In this way, some parameters can be modified by someone who doesn't know how to use Rapid Miner, but knows what kind of result to expect after the process execution. Other advantage is that the system administrator can execute the process and check the result anytime and anywhere, using any device with Internet access and a browser installed, without the need to use the computing system where the database and Rapid Miner are installed.

References

[1] Christopher Clifton, *Encyclopedia Britannica: Definition of Data Mining*, https://www.britannica.com/technology/data-mining
[2] Aguiar-Pulido V., Seoane J.A., Gestal M., Dorado J., *Exploring patterns of epigenetic information with data mining techniques*, Curr Pharm Des. 2013; 19(4):779-89
[3] Kevin Roebuck, *Data Mining: High-Impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*, Lightning Source, 2011, ISBN 1743047029, 9781743047026
[4] Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques, 2nd 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
[5] Mircea-Adrian Mușan, Ionela Maniu, *Extracting Associations Rules with FP-Growth and Apriori from Commercial Transactions*, "OVIDIUS" UNIVERSITY ANNALS – CONSTANTZA Year XVI – Issue 16 (2014) Series: CIVIL ENGINEERING
[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 classes documentation: http://fossies.org/linux/rapidminer/javadoc/overview-summary.html
[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] Ch. Borgelt, *Frequent Pattern Mining*, Intelligent Data Analysis and Graphical Models Research Unit European Center for Soft Computing, 33600, Mieres, Spain, 2005

Mircea-Adrian MUȘAN
"Lucian Blaga" University of Sibiu
Department of Informatics
Sibiu, Street Ion Rațiu, No. 5
ROMANIA
E-mail: mircea.musan@ulbsibiu.ro

Iuliana-Maria CÂNDEA
"Lucian Blaga" University of Sibiu
Department of Informatics
Sibiu, Street Ion Rațiu, No. 5
ROMANIA
E-mail: iuliana.maria.candea@gmail.com