

An Agent-Based Approach for Extracting Business Association Rules from Centralized Databases Systems

Nadjib Mesbahi, Merouane Zoubeidi, Abdelhak Merizig, Okba Kazar
LINFILaboratory, Computer Science Department, University of Biskra
07000, Biskra, Algeria
{najib.mes@gmail.com} {mzoubeidi01@gmail.com} {a.merizig@univ-biskra.dz}
{o.kazar@univ-biskra.dz}



Journal of Digital
Information Management

ABSTRACT: Today, enterprises use a variety of applications to manage day-by-day business activities using a large centralized database. Since a huge amount of data stored in this centralized database produced by the daily use of several systems, it is important to integrate decision-making tools to analyse and interpret these business data. For this purpose, Data Mining is a powerful technology that promote information and knowledge extraction from large databases. In this paper, we present an agent-based approach for extracting business association rules from centralized database systems. This approach combine paradigm of multi-agent system and the association rules as a data mining technique to build an efficient model. It is relying on the intelligent partitioning of data to make the execution of business association rules in a parallel and distributed way from a large centralized database. To validate our approach, we applied it during the realization of a real case study on ERP database at the National company of Well Services (ENSP) using JADE platform with machine learning WEKA toolbox for association rules mining. The developed system has been compared with the classic association rules algorithms and has proved it is more efficient and more scalable. The main objective of our work is to improve and accelerate the process of extracting association rules by business through centralized database systems. As a result, the decision process of these systems becomes more improved.

Subject Categories and Descriptors: [I.2.11 Distributed Artificial Intelligence]: Multiagent systems; [H.2.8 Database Applications]: Data mining; [J.1 ADMINISTRATIVE DATA PROCESSING]: Business

General Terms: Business Association, Centralized Databases, Databases, Data Mining, Association Rules, Agent-based approaches, Multi-agent Systems

Keywords: Centralized Database Systems, Multi-Agents System (MAS), Data Mining, Business Associate Rules, ERP, JADE, WEKA

Received: 11 March 2019, Revised 2 June 2019, Accepted 22 June 2019

Review Metrics: Review Scale- 0/6, Review Score-4.15, Inter-reviewer Consistency- 76%

DOI: 10.6025/jdim/2019/17/5/270-288

1. Introduction

Enterprises data are increasing daily due to their operations that include data processing in every task. The large data getting bigger the analyze or extracting the relevant information will be an issue in data management. Nowadays, enterprises and companies use different sites or agency everywhere to gain more clients and more works. Therefore, these sites oblige the owners to dispatch their data in each site. Managers look always for a solution that helps them to make decisions in their work. To extract relevant data that might help the managers for a specific objective from huge represents a critic task due to the complexity and time to achieve the results. This amount of data could hide some part of knowledge that can affect the final decision. Data mining is one of the technics that can help to explore and extract specific knowledge from

databases (Conrad et al., 2016). In addition, these technics are integrated into centralized database systems that aim to enhance the information extraction by a new knowledge to support the managers in the process of making a decision. Furthermore, proposing an architecture that proposes decisions can help the companies to exclude extraneous ones.

Decision making is one of the major problems confront the enterprise's owner especially when they need to respond quickly for urgent situations. In order to make a decision, the managers look forward to automatic materials to help them make decisions to reach their objectives. However, centralized database systems need more work because it is getting bigger and it is increased exponentially for each use (Zoubeidi et al., 2019). In fact, to run a data mining algorithm on centralized database systems it will suffer from processing resources that cover memory and processors, especially when business data is increased in its database. Due to a large amount of data, this could hide knowledge in data lead to false decisions which can affect the enterprise's funds. Furthermore, as business data are increasing it is necessary to use some method that can extract the relevant one. In this paper, we use association rules mining technic which stands on three phases to come up with these rules (i) discovering a set of frequent itemsets, and (ii) generating the association rules and (iii) generate all the confident association rules (Pierrard et al., 2018). To achieve a better solution to the mentioned problem is consists to minimize the response time used to process the algorithm. In this scope, many works in literature have used the different paradigm to afford the data mining algorithms more effective (Albashiri et al., 2010), (Coenen et al., 2006), (Hamza et al., 2017) and (Pudi et al., 2003). Nowadays, the integration of the multi-agent system paradigm in data exploration process is considered as a powerful and effective technology in term of data treatment complexity (Mesbahi et al., 2018). Using the multi-agent paradigm could enhance in different perspective like simulating complex systems (Merizig et al., 2019). Moreover, these data are distributed in a way that the data will be separated over a set of communicating entities, autonomous and with reactive skills these entities represented by the agents. (Mesbahi et al., 2015).

This paper is organized as follows: Section 2 outlines some works related to our problem. In section 3 we present our approach and we provide the different components of our proposal. In addition, it presents some algorithms used by each agent. Section 4 shows the implementation and some results using our proposed model followed by discussion. Finally, section 5 concludes this paper.

2. Related Work

In the literature of data mining, there are many algorithms have been proposed for DARM (Distributed Association Rule Mining). Actually, the exponential size of databases makes association rules extraction more difficult in one

process, which requires a resource-intensive processor and very large memory to finalize their treatments. Therefore, it is necessary for researchers to propose efficient methods to carry out these mining tasks. One of the solutions to this problem is the distribution of huge database on small partitions, which can enhance the extraction of association rules at the right time.

Many approaches are proposed for data partitioning like the work of (Albashiri et al., 2010), (Coenen et al., 2006), and (Pudiet al., 2003). DARM algorithms proposed in many works, these ones tried to reduce the number of false Itemsets candidates. These works considered the priority of data partitioning as a principal factor to enhance the treatment process in data mining. In the literature, there are two types of data partitioning: (i) horizontal partitioning and (ii) vertical partitioning. (Albashiri, 2010) and (Albashiri et al., 2010), their works interested in horizontal partitioning and for the second type that used the vertical partitioning given in works (Albashiri, 2010) and (Connell et al., 2004). In fact, there are three categories proposed by many works like (Albashiri, 2014) and (Zaki, 2000) for DARM algorithms which uses the Apriori algorithm and their variants. These categories are: Data Distribution, Candidate Distribution and Count Distribution (Cheung, et al., 1999), (Agrawal et al., 1996) and (Deypir et al., 2009). Currently, data partitioning has become possible in many contexts by the use of Multi-agents system. In the literature, there are many approaches that use multi-agents systems to solve problems especially that the ones that need the distributed elements and autonomy aspects to achieve the principal goal (Ferber, 1999) and (Fiolet, 2006). Since this paradigm has many benefits, multi-agents systems used as a powerful technology to develop a system that can handle the extraction of distributed associations rules (DARM) (Symeonidis A., 2007) and (Zhang, et al., 2003). The remainder of this section we are going to present three works that solve distributed association rules extraction (DARM) using multi-agents system.

(Albashiri, 2014) have proposed a data distributed approach based on agents in order to extract the association rules in two forms: parallel and specified distribution based on data partitioning (vertical and horizontal). In order to facilitate the partitioning operation, tree data structure is used Apriori-T with association rules algorithm. Measures comparison shows that described data partitioning methods are extremely effectively to limit the memory requirement during the execution of association rules algorithms, thus execution time change slowly and linearly according to the augmentation of data dimension.

Additionally, in the work of (Ogunde et al., 2015), an approach PEMA (Partition Enhanced Mining Algorithm) has proposed for data partitioning in order to enhance the extraction of distributed association rules (DARM). Extraction process in this approach contains two steps: the first step data agents segment the database

horizontally in small partitions according to available sites and size of memory. The second step if the database is huge (big number of columns), will partition it vertically. After this, mobile agents that use association rules to discover the local frequent Itemsets. In addition, the local's frequent Itemsets are integrated incrementally from data site to another in order to obtain the global frequent Itemsets. The obtained experimental results from real data prove that the average time response of PEMA gives enhanced results compared to existing algorithms. In addition, PEMA uses an inferior cost of communication with the small average size of exchanged messages compared to existing DARM systems.

Furthermore, the authors of (Ogunde et al., 2011), have proposed theoretical cost models for DARM based on agents used in distributed data with prototype model entitled MADARM (Multi-agent Distributed Association Rule Miner). These cost models help the based models to estimate and to predict time response of DARM task. Many agents are integrated in this system like Association Rule Mining Coordination Agent (ARMCA) to perform and to coordinate with the other agents, Mobile Agent-Based Association Rule Miner (MAARM) to execute ARM task in each distributed data source. Mobile Agent-Based Result Reporter (MARR) created by MAARM agent for the results migration to ARMCA and results integration coordinator agent (RICA) founded to integrate knowledge and results. Knowledge integration is optimized on data sources using distributed agent-based knowledge integration (ADKDI). In this study, Apriori and FP-Growth algorithms are used to extract local frequent itemsets. Only the conceptual viewpoints are presented in this work while the researchers concluded that this work still needs some improvement and experimental validation.

In this purpose, we propose a distributed association rules approach by business from centralized databases systems, using the Apriori algorithm. Our proposed approach based on multi agents system and horizontal data partitioning. The use of Multi Agents System (MAS) is very important in our work because it allows for the creation of clear and robust knowledge models by integrating cooperative agents. Using this paradigm into the process of extracting association rules, could add some benefits such as relying on distribution, collaboration, flexibility, the reuse, scalability and efficiency of multi-agent systems (Zoubeidi et al., 2019). Furthermore, our proposed approach based on a distributed architecture which equally capable of running multiple association rule extraction processes in a parallel way and around a centralized database using a Apriori algorithm and horizontal partitioning. This reduces the response time and the cost of communication in our system.

3. Proposed Approach

Data relevant extraction in distributed databases through different sites it is a big challenge to companies and enterprises. Due to the number of operations and

transactions created every day which makes the processing and the interpretation more difficult. In order to help managers and decision makers to come up with the solution our proposal covers most of the problems in the enterprises.

The aim of our work is to handle some points given as follows:

1. Reduce overall time consumed by extracting association rules.
2. Improve the quality of knowledge presentation to data mining user, by using the clearer rules offered by the association rules technique.
3. Partitioning the business data used in datamining process on smaller pieces of data to lighten the general process of datamining.
4. Distributing these partitions of data on multiple remote machines to make parallel and distributed data processing that will reduce the overall time of DM process execution.
5. Horizontal partitioning of data makes it possible to produce more consistent association rules.

Moreover, our approach is based on a distributed architecture, which able to make the execution of the association rules extraction process in parallel around a huge database. It allows data to be partitioned by the business on smaller data partitions. After that, an association rules agent on a different machine will treat the data of each job separately.

3.1 Architecture Description

In this subsection, we explain the different components used in our Datamining associate rule extraction with their roles. Our architecture aims to solve the problem of relevant data extraction from an amount of data in enterprises in order to help managers in the decision making the process. As mentioned before this architecture composed of three layers to handle the whole operation of data extraction. Thus, Multi agent systems could enhance the processing task using its behavior and its proprieties.

The general architecture proposed in (Figure. 1) which is divided into layers, each layer has its own features with specific responsibilities. The detailed description of the different layers and their relationships given as follows.

1. End-User layer: It is the interface of Knowledge Discovery Layer and with the centralized databases systems in particular, ERP (Enterprise Resources Planning), CRM (Customer Relationship Management) and SCM (Supply Chain Management). Its main role is to capture the purpose of the users in order to meet their best needs. The systems interface interacts with ERP users, CRM users or SCM users to help them perform a specific task in their business. The DM interface agent interacts with the datamining user to assist in the formulation of datamining requests such as minimum media capture and minimal confidence as well as the

accuracy of the data path made datamining object.

2. Data Layer: It is an intermediate layer, composed of a set of business and a single and central database, having a variety of data collected from any business. This layer also plays a fundamental role in communication with the end-user layer to respond to user's requests and the knowledge discovery layer from the centralized database.

3. Knowledge Discovery Layer: This layer models the entire process of data mining, which goes through a set of successive steps given as follows: Data Selection, Data Cleanup...etc. Its main purpose is to discover new useful and unknown knowledge hidden in the centralized database. This new knowledge is represented in the form of business association rules to help decision-makers

making the right decisions at the right time.

3.2. Agent based Approach for Business Extraction

In this subsection we describe the role of each component that used in our system and presented in (figure. 1) in order to extract the relevant data using agent's paradigm. After preparing and transforming the business data as datamining into the Mining-Oriented Data Base (MODB) (Figure 1), a very important step devoted to modelling the datamining stage based on association rules by the Multi-Agent System paradigm (Zoubeidi et al., 2019). In our MAS architecture illustrated in (Figure 2.), each cognitive agent has a particular skill used to solve part of the overall problem of extracting association rules by business from a large and centralized database.

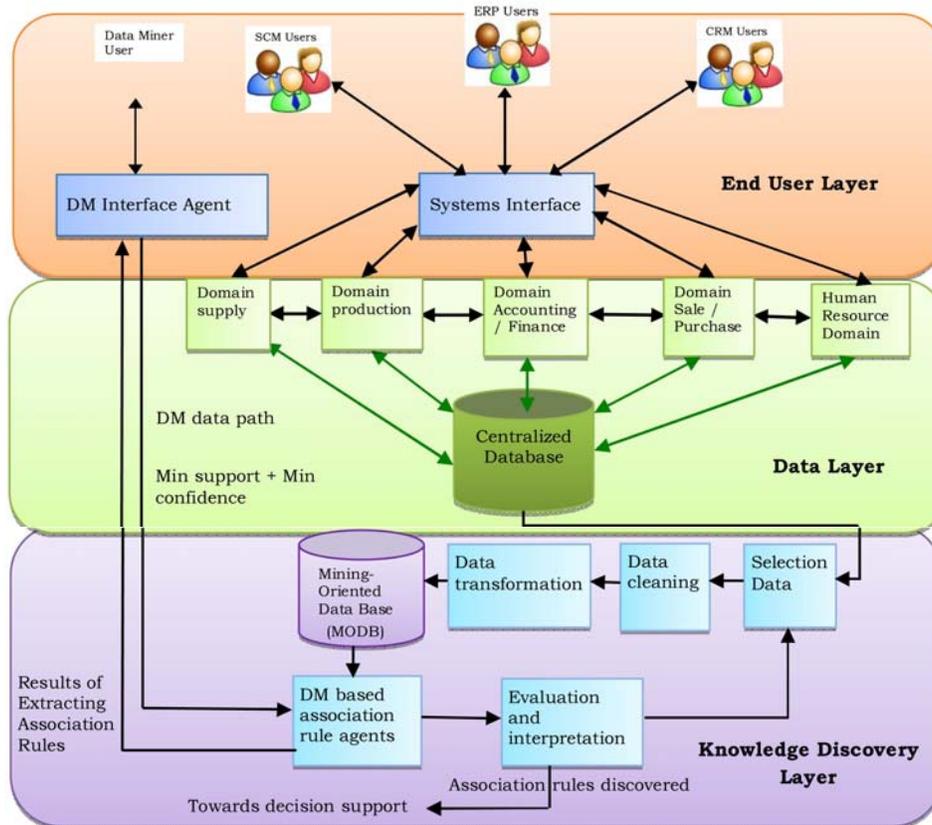


Figure 1. Data Mining architecture based agent from centralized databases systems

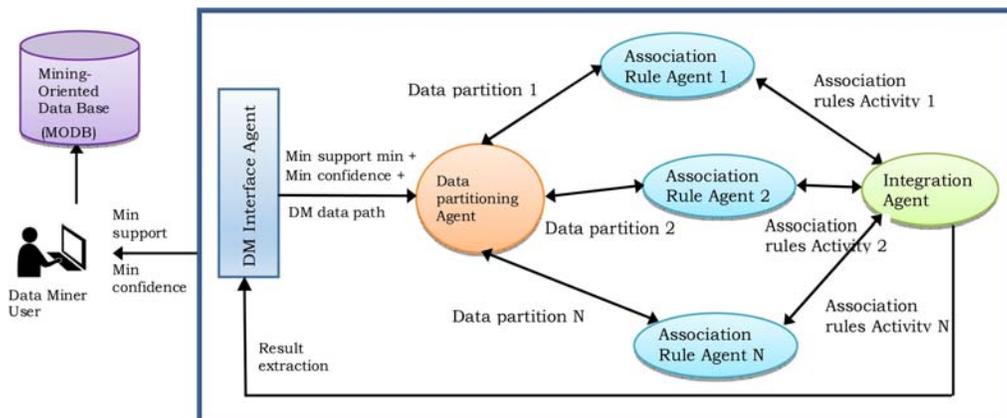


Figure 2. Architecture of the Multi-Agent System

The main advantage of the proposed MAS architecture lies in the ability to optimize execution of the association rule algorithm around a large database while basing on distribution, flexibility, reuse, scalability and efficiency of multi-agent systems (Zoubeidi et al., 2019). Indeed, the DM interface agent provides the necessary parameters to launch the MAS architecture. Subsequently, the data partitioning agent horizontally partitions the huge amount of data per business, in the question of Data Mining, on smaller subsets of data (partitions) according to a criterion introduced by the datamining user. This partitioning is important, the purpose of which is to distribute the processing load of the association rules algorithm over several sub-processes working in parallel distributed, on small data partitions and on several remote machines. Then, the data filtering agent creates in parallel with scalability, a set of association rule agents according to the number of data partitions. Then, the association rules agents will run in parallel for the generation of association rules by the business. After that, the interpreting agent, in turn, will receive all of the results of the association rules for each business and send them as a final result of extracting business association rules to the Data Miner user for the business. Display and final operation. The main goal of our MAS architecture is the lacing of several association rule extraction processes simultaneously in order to reduce the overall time consumed by the generation of association rules on the centralized database, the agents used in our MAS architecture are cognitive types, since they are able to predict and anticipate actions.

To summarize, our proposed MAS architecture is adapted to the context and the specificities of our system of extraction of the rules of association by business that makes it possible in particular to:

- Adaptive partitioning using a horizontal division of data by business,

- Efficiency by the distributed parallel launch of multiple instances the association rules algorithm and
- Reliability by presenting clear results in the form of association rules (by type of business) in the favor of the data-mining user.

1) DM Interface Agent

The DM interface agent is the only agent that interacts at the same time, with the Data Miner user and with the DM system. It provides a graphical interface to interact with the DM system, it allows the Data Miner user to formulate his datamining queries; to enter the minimum support and minimal confidence, as well as the data path, made the object of knowledge extraction. Then it passes the parameters entered to the Data partitioning agent to start the data splitting operation by the business. Finally, this agent receives the results of the extraction of the association rules to present them to the Data Miner user to help for the decision. The user Interface Agent goes through the following steps (Figure 3) to achieve its goals (Zoubeidi et al., 2019).

In Figure 4. we display the pseudo code of Data Mining Interface agent (Zoubeidi et al., 2019).

2) Data Partitioning Agent

It is a cognitive agent whose role is the partitioning of business data, doing the object of the extraction operation of association rules. It decomposed horizontally the huge mass of data by business on several large month data partitions. After that, it launches in parallel, several instances of the association rules agent by profession to speed up the process of extracting association rules. The Data partitioning agent follows the following steps in Figure 5 to complete its tasks (Zoubeidi et al., 2019).

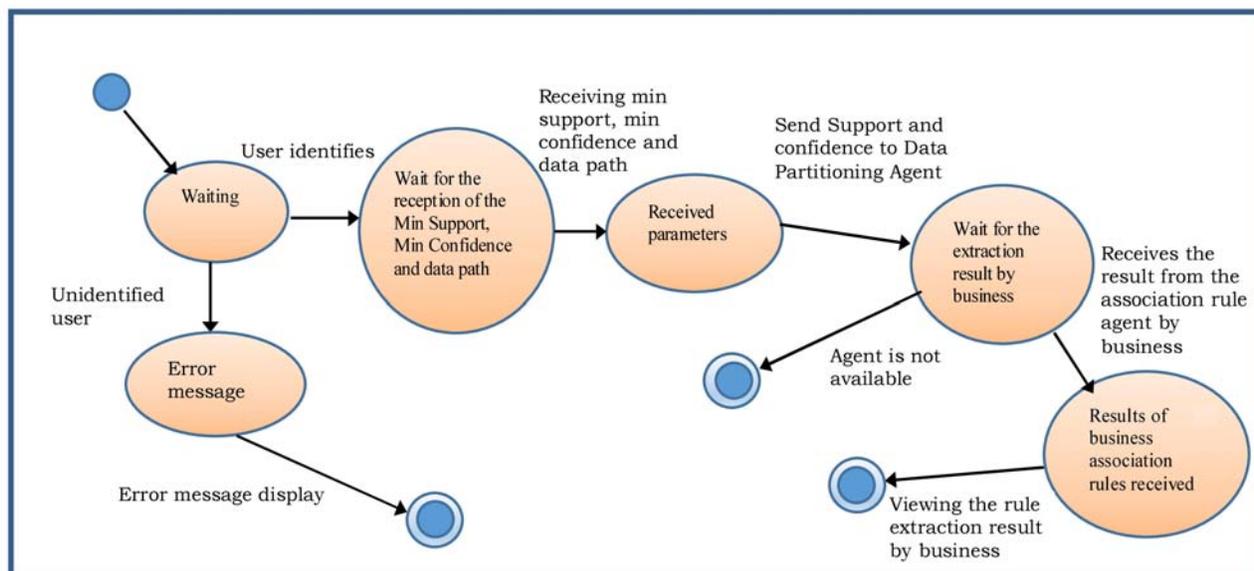


Figure 3. Status Diagram -Data Mining Interface Agent

```

Input: Support min, confidence min, path of MODB;
Output: Extraction association rules by business results;
Begin
Wait all;
// wait for minimum support and minimum confidence from Datamining user;
/*first execution of DM interface agent */
Create the Data Partitioning Agent;
Compute; /* used to compute the number of association rules results by the association rules
agents*/
/*Num_Partition: the number of business partitions */
/*If the association rule agent finish its work, then will send it to DM interface agent; this one will
compute the number of received results. If compute = Num_Partition then it will display the results
of the extraction of association rule by business */
If Compte == Num_Partition then
    Display the results of the extraction of association rule by business;
End if

```

Figure 4. Displaying pseudo-code of Data Mining Interface Agent

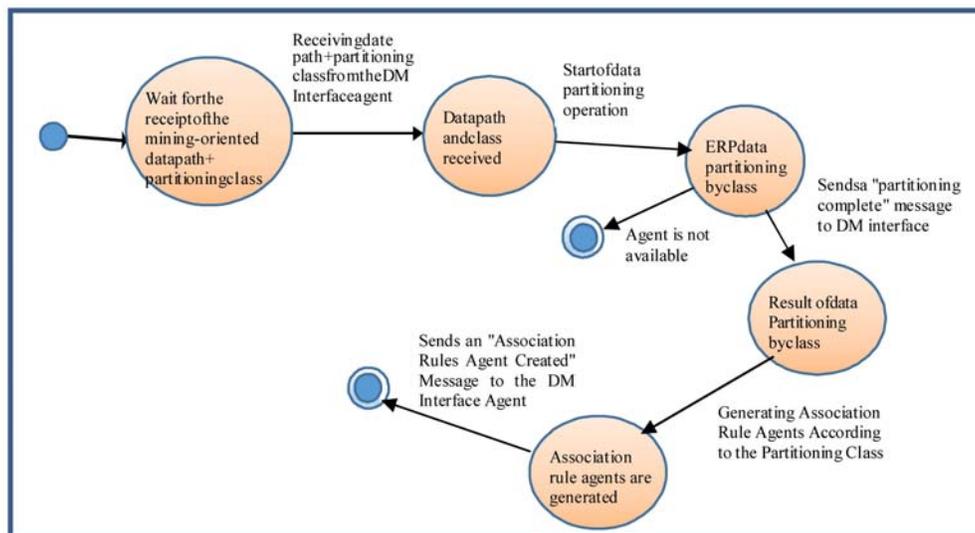


Figure 5. State diagram of the Data Partitioning Agent

```

Input: Mining-oriented database (MODB);
Output: Partitions of data partitioning by business;
Begin
Wait all; /*wait for MODB path from DM interface agent. */
i= 1; /* i used for browse chosen Partition by business (1 >= i <= Num_Partition)
/*Num_Partition: it's the number of Partitions by business */
/* X: it's the line in MODB database
ForEach X ∈ MODB do
    ForEach i <= Num_Partition do
        If (Xi ∈ Partition_i) then
            Assign Xi in partition business_i;
        Endif
    EndFor
EndFor
Send the data of the last partitioning operation to DM interface agent;
i= 1;
ForEach i <= Num_Partition do
    Create an agent for association rule i;
    Send partition_i to association rule agent i;
    Send message "Agent association rule i" created and initialized;
EndFor

```

Figure 6. Displaying Data partitioning agent pseudo code

In Figure 6. we display the pseudo code of data partitioning agent (Zoubeidi et al., 2019).

3) Associate Rule Agent

Multiple instances of the association rule agent are dynamically and scalable created by the data partitioning agent, depending on the number of business data partitions. Its role is the execution of the association rules extraction algorithm after the receipt of the minimum support and minimal confidence. Also, another role of this agent is sending the results of association rule extraction to DM user interface agent for display and interpretation. This agent follows the following steps (Figure 7) to complete its tasks (Zoubeidi et al., 2019).

In Figure 8 we display the pseudo code of association rules agent (Zoubeidi et al., 2019).

3.3 DARM Scenarios in the Proposed Architecture

In our architecture, the used agents are interface agent DM, Data partitioning agent, association rules extraction agent, integration agent. These agents are created using

JADE environment and they are registered with the Registrar (RA). The DARM process is starting by the interface agent DM that provides the interface between the system and the Data miner user to respond to requests from the system. This agent will receive the three parameters necessary to start the DARM process, which are the minimum support, the minimum confidence and the path where data is located for the sends to the Data Partitioning Agent. The latter, in turn, breaks down the data horizontally and is the subject to knowledge extraction on several data partitions by the business. After that, it launches a set of association rule agents that are required to perform DARM tasks based on the number of business partitions. The association rule agents are responsible for performing the ARM task in a distributed parallel fashion to generate the association rules by the business. Once one of the knowledge extraction operations is completed by one of the association rule agents, the result sent directly to the results integration agent. This in turn will send the result of the business association rule to the DM interface agent for display and final presentation. This description illustrated in Figure 9 (Zoubeidi et al., 2019).

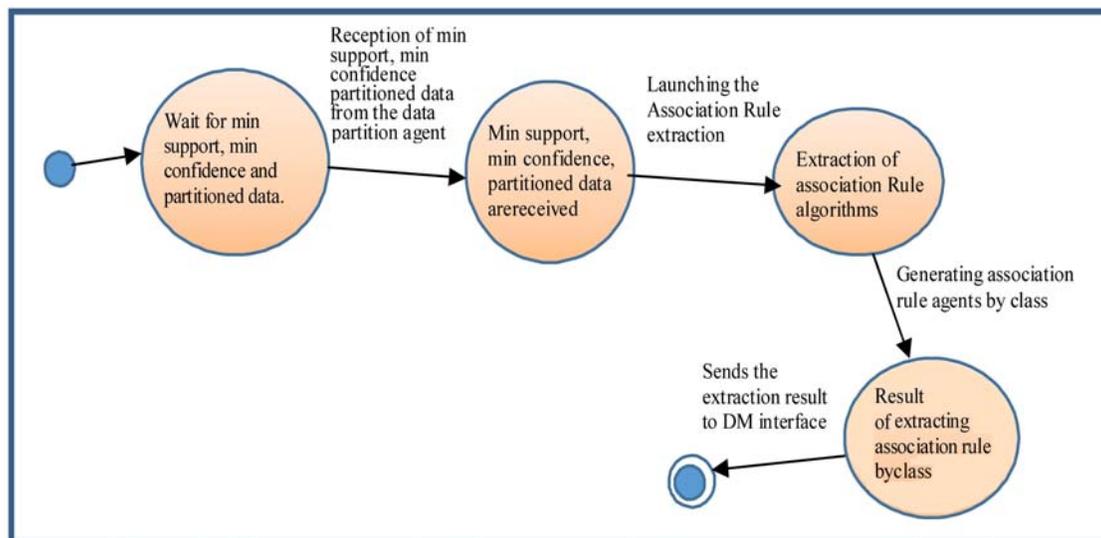


Figure 7. Shows the status diagram of the association rules agent

```

Input: min Support, min confidence, data partitioning by business;
Output: Association rules by business;
Begin
Wait all; // wait for minimum support and minimum confidence from DM interface agent and
the data partitioning by business from data partitioning agent.
i= 1; // i used for browse the chosen class (1 >= i <= Num Partition)
/*Num_Partition: It's the number of business partitions */
Receive the min support and confidence from DM interface agent;
Receive partition i of data partitioning by business from data partitioning agent;
Starting association rule agent i;
Send association rules by business to DM interface agent to display results;
Notify the end of the rules association i extraction operation to DM interface agent;
End
  
```

Figure 8. Displaying pseudo code association rules agent

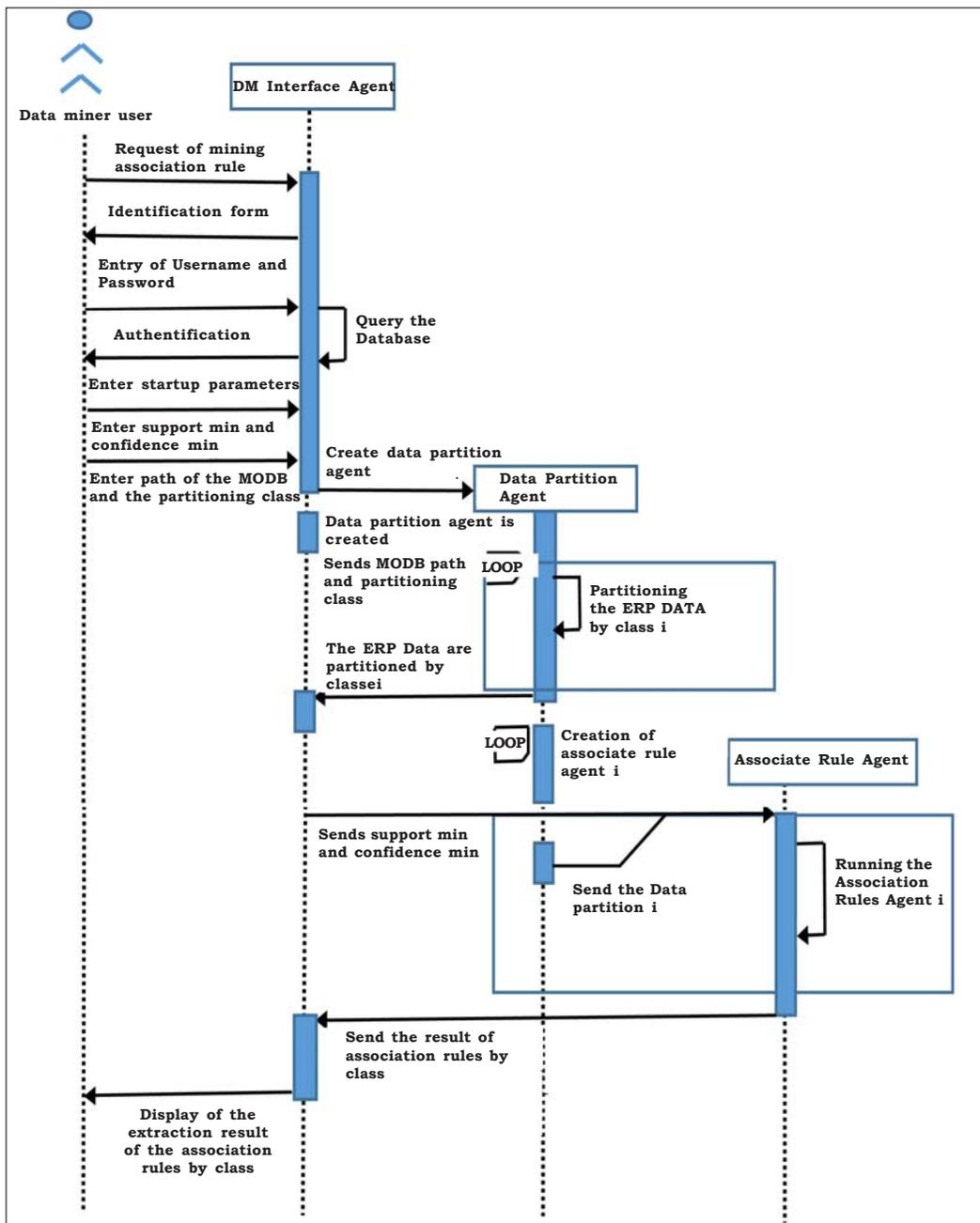


Figure 9. AUML sequence diagram

4. System Implementation and Results

In our experimentation, as a case study, we have the ERP Well Services Database, which is now in the testing phase, of the National Well Service Company (ENSP).

4.1 Field Description of the Application

Day after day and with daily use, the ERP database of ENSP Company accumulated a huge amount of the well data services, which hide decisive knowledge about the market and competition, but remain little exploited. To meet this need, we have proposed the use of an association rule technique to analyze and look for well services that tend to be realized together. Association rules allow analyzing customer behavior and understanding their service delivery patterns in order to establish effective

NPMS strategies, such as:

1. Preparation and organization of the equipment necessary to carry out such a service to the wells;
2. Know in advance the well services that are provided in package (together);
3. Selection and optimization of the personnel necessary for the realization of professional services (Pumping, Texteam, Snubbing ... etc);
4. Optimization of inventory management and store shelf layout.
5. Prediction of future service requirements of customer by business line to anticipate necessary actions.

4.2 Implementation Tools

In order to develop the proposed approach, many tools are used. Java Eclipse IDE for Java Developers was used as a development environment. JADE 3.8 has been used as a multi-agent system development middleware platform. The WEKA tool is an open-source self-learning toolkit written in JAVA, has been used for knowledge extraction in our approach. The VMware Workstation 12 Pro was used to simulate ERP data distribution in a distributed environment. The Postgres Database of ERP Odoo was used as a test base to test the developed system.

• The JADE Platform

JADE (Java Agent Development Framework) is an agent development environment fully implemented in the JAVA language. He facilitates the implement of a multi-agent system and attempts to optimize distributed agent system performance while meeting Foundation for Intelligent Physical Agents (FIPA) specifications across a set of components. YUAN., T. (2008)

• WEKA Tool

WEKA is an open-source Toolkit for knowledge extraction, written in java, developed at Waikato University, New Zealand. It is now used in many different fields, especially research and education. In our work, we used the WEKA platform as a tool for ERP data preprocessors and as a library for association rule algorithms to integrate with Jade developed agents.

4.3 Data Preparation

The data used in our experiment are exported from the ERP-Postgres database on ENSP services. These data represent an important mass of well services data, hiding information and knowledge that is useful for making strategic business decisions. In this phase, ERP data extracted is obtained from an Excel export, a set of tables related to well services. These tables are Invoice, Invoice lines, Service Lines and Activity that are presented later:

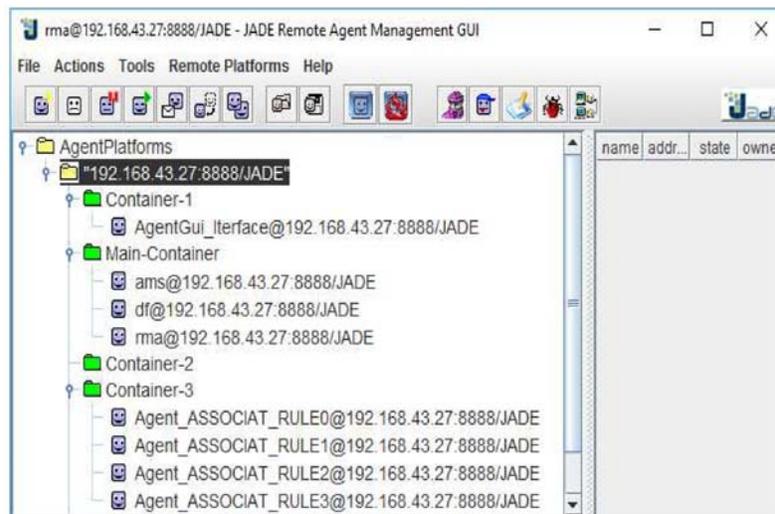


Figure 10. Agents of our system developed with JADE

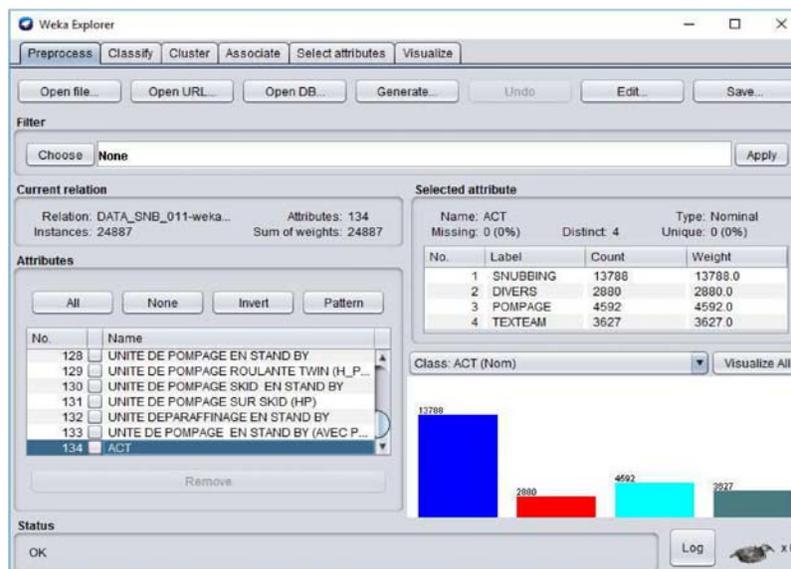


Figure 11. Well Services of the ENSP-Snubbing Directorate under Weka 3.8

	A	B	C	D
	IDFACTURE	Code Service Pu		Quantite
2	2,2007,1	113000	399936	2
3	2,2008,1	112000	33328	8
4	2,2009,1	103000	469274	11
5	2,2010,1	102000	39106	12
6	2,2011,1	103000	469274	11
7	2,2012,1	101000	938548	1
8	2,2013,1	102000	39106	9
9	2,2014,1	103000	469274	13
10	2,2015,1	101000	938548	1
11	2,2017,1	113000	399936	19
12	2,2007,2	112000	33328	7
13	2,2008,2	111000	799872	1
14	2,2009,2	112000	33328	6
15	2,2010,2	113000	399936	4
16	2,2011,2	140000	110000	6
17	2,2012,2	140000	110000	20
18	2,2013,2	101000	938548	1
19	2,2014,2	102000	39106	5
20	2,2015,2	500000	45000	7
21	2,2016,2	581100	10080	93
22	2,2007,3	102000	39106	3
23	2,2008,3	101000	938548	1
24	2,2009,3	581100	10080	31
25	2,2010,3	130006	28600	4
26	2,2011,3	130005	22800	24
27	2,2012,3	130003	200	13
28	2,2013,3	130002	121100	31
29	2,2014,3	130002	1953	31
30	2,2015,3	581100	10080	3
31	2,2016,3	581100	10080	31
32	2,2007,4	112000	33328	6
33	2,2008,4	111000	799872	1

Table 1. Service Lines

	A	B	C	D	E	G	H	I	J
	IDFACTURE	Nfact	Code Client	Date	Facture ACT	Lservice	month	net	Année
2	2,2007,1	1	17150	31/01/2007	01	21	14226503,1	14226503,1	2007
3	2,2008,1	1	17150	31/01/2008	01	02	410795,67	410795,67	2008
4	2,2009,1	1	17150	31/01/2009	02	02	1066496	1066496	2009
5	2,2010,1	1	17311	28/01/2010	04	11	3351931,87	3351931,87	2010
6	2,2011,1	1	17001	31/01/2011	03	07	10300124,6	10300124,6	2011
7	2,2012,1	1	17001	31/01/2012	03	08	831656,73	831656,73	2012
8	2,2013,1	1	17001	31/01/2013	01	07	3842827,71	3842827,71	2013
9	2,2014,1	1	17001	31/01/2014	01	07	5506141,28	5506141,28	2014
10	2,2015,1	1	17001	18/01/2015	02	12	-10843826,2	-10843826,2	2015
11	2,2017,1	1	17150	31/01/2016	04	08	2983673,9	2983673,9	2016
12	2,2007,2	2	17150	29/01/2007	03	12	7005147,42	7005147,42	2007
13	2,2008,2	2	17150	31/01/2008	03	02	2183703,35	2183703,35	2008
14	2,2009,2	2	17150	31/01/2009	01	02	7415480	7415480	2009
15	2,2010,2	2	17311	31/01/2010	01	11	8367415,12	8367415,12	2010
16	2,2011,2	2	17001	31/01/2011	02	07	9242927,01	9242927,01	2011
17	2,2012,2	2	17001	31/01/2012	04	08	10954926,8	10954926,8	2012
18	2,2013,2	2	17001	31/01/2013	03	07	2839103,98	2839103,98	2013
19	2,2014,2	2	17001	31/01/2014	03	07	11757905,7	11757905,7	2014
20	2,2015,2	2	17001	18/01/2015	01	12	10541769,7	10541769,7	2015
21	2,2016,2	2	17150	31/01/2016	01	08	5253860,54	5253860,54	2016
22	2,2007,3	3	17150	31/01/2007	02	12	7567288,89	7567288,89	2007
23	2,2008,3	3	17150	31/01/2008	04	02	3351227,92	3351227,92	2008
24	2,2009,3	3	17150	31/01/2009	03	13	7598784	7598784	2009
25	2,2010,3	3	17150	28/01/2010	03	02	4454287,2	4454287,2	2010
26	2,2011,3	3	17001	31/01/2011	01	08	1751927,35	1751927,35	2011
27	2,2012,3	3	17001	31/01/2012	01	08	7198132,56	7198132,56	2012
28	2,2013,3	3	17001	31/01/2013	02	07	8087144,93	8087144,93	2013

Table 2. Invoice

	A	B	C
	N° Enr.	code act	Design
2	1	01	SNUBBIN
3	2	02	POMPAGE
4	3	03	TEXTTEAM
5	4	04	DIVERS

Table 3. Activity

ID	Description
100103	TARIF HORAIRE HRL 120 K
100104	MOB.OU DEMOB. DE REGION A REGION (100 A 300K)M
100105	MOB OU DEMOB DE REGION A REGION (300 A 500 KMS
100106	MOB OU DEMOB. DE REGION A REGION (PLUS DE 500 KI
100108	TARIF HORAIRE STAND BY HRL 120 K
100200	MOBILISATION UNITE HRS 150 K (AU DELA DE 300 KMS)
100201	DTM DE PUIITS A PUIITS HRS 150 K
100202	TARIF JOURNALIER HRS 150K
100203	TARIF HORAIRE HRS 150 K
100207	DEMOBILISATION (300 A 500 KMS)
100208	TARIF HORAIRE STAND BY HR 150 K
100209	MOBILISATION (100 A 300 KMS) HRS 120K/150K
100210	DEMOBILISATION (100 A 300 KMS) HRS 120K/150K
100211	MOBILISATION ENTRE REGIONS PLUS DE 500 KMS
100212	DEMOBILISATION ENTRE REGIONS (DE 300 A 500 KMS)
100300	MOBILISATION HRS 225 / BHS 235 DE REGION A REGION
100301	DTM DE PUIITS A PUIITS HRS 200K A 225K/BHS 235
100302	TARIF JOURNALIER HRS 200 K A 225 K / BHS 235
100303	TARIF HORAIRE HRS 200 K A 225 K / BHS 235
100304	STAND BY JOURNALIER HRS 200 K A 225 K / BH
100305	STAND BY HORAIRE HRS 200 K A 225 K / BHS 235
100400	MOBILISATION UNITE SNUBBING HRS 340 (AU DELA DE 5
100401	DTM DE PUIITS A PUIITS HRS 340 K
100402	TARIF JOURNALIER HRS 340 K
100403	TARIF HORAIRE HRS 340 K
100404	MOB. OU DEMOB. DE REGION A REGION 0 (100 A 300 KM
100405	MOB OU DEMOB (REGION A REGION) (DE 3 00 A 500 K
100406	MOB. OU DEMOB. DE REGION A REGION DE (PLUS DE 5
100408	DTM DE PUIITS A PUIITS (AU DELA DE 5 KMS)
100600	MOBILISATION (MONTAGE ET TEST INCLUS)

Table 4. Well Services Linen

ID	IDFACTURE	Nfact	Code_Client	Date	Facture ACT	Montht	net	Année	LinService	PU
2	2.2007.1	1	17150	31/01/2007	SNUBBING	14226503,1	14226503,1	2007	MOBILISATION UNITE HRS 150 K (AU DE	39106,00
3	2.2008.1	1	17150	31/01/2008	SNUBBING	410795,67	410795,67	2008	DTM DE PUIITS A PUIITS HRS 150 K	469274,00
4	2.2009.1	1	17150	31/01/2009	POMPAGE	1066496	1066496	2009	TARIF JOURNALIER HRS 150K	938548,00
5	2.2010.1	1	17311	28/01/2010	DIVERS	3351931,87	3351931,87	2010	TARIF HORAIRE HRS 150 K	39106,00
6	2.2011.1	1	17001	31/01/2011	TEXTEAM	10300124,6	10300124,6	2011	DEMOBILISATION (300 A 500 KMS)	469274,00
7	2.2012.1	1	17001	31/01/2012	TEXTEAM	831656,73	831656,73	2012	TARIF HORAIRE STAND BY HR 150 K	938548,00
8	2.2013.1	1	17001	31/01/2013	SNUBBING	3842827,71	3842827,71	2013	MOBILISATION (100 A 300 KMS) HRS 12	399936,00
9	2.2014.1	1	17001	31/01/2014	SNUBBING	5506141,28	5506141,28	2014	DEMOBILISATION (100 A 300 KMS) HRS	33328,00
10	2.2015.1	1	17001	18/01/2015	POMPAGE	-10843826,2	-10843826,2	2015	MOBILISATION ENTRE REGIONS PLUS C	799872,00
11	2.2017.1	1	17150	31/01/2016	DIVERS	2983673,9	2983673,9	2016	DEMOBILISATION ENTRE REGIONS (DE	33328,00
12	2.2007.2	2	17150	29/01/2007	TEXTEAM	7005147,42	7005147,42	2007	MOBILISATION HRS 225 / BHS 235 DE F	399936,00
13	2.2008.2	2	17150	31/01/2008	TEXTEAM	2183703,35	2183703,35	2008	DTM DE PUIITS A PUIITS HRS 200K A 225I	110000,00
14	2.2009.2	2	17150	31/01/2009	SNUBBING	7415480	7415480	2009	TARIF JOURNALIER HRS 200 K A 225 K	110000,00
15	2.2010.2	2	17311	31/01/2010	SNUBBING	8367415,12	8367415,12	2010	TARIF HORAIRE HRS 200 K A 225 K	938548,00
16	2.2011.2	2	17001	31/01/2011	POMPAGE	9242927,01	9242927,01	2011	STAND BY JOURNALIER HRS 200 K A ;	39106,00
17	2.2012.2	2	17001	31/01/2012	DIVERS	10954926,8	10954926,8	2012	STAND BY HORAIRE HRS 200 K A 225 K	45000,00
18	2.2013.2	2	17001	31/01/2013	TEXTEAM	2839103,98	2839103,98	2013	MOBILISATION UNITE SNUBBING HRS 3	10080,00
19	2.2014.2	2	17001	31/01/2014	TEXTEAM	11757905,7	11757905,7	2014	DTM DE PUIITS A PUIITS HRS 340 K	39106,00
20	2.2015.2	2	17001	18/01/2015	SNUBBING	10541769,7	10541769,7	2015	TARIF JOURNALIER HRS 340 K	938548,00
21	2.2016.2	2	17150	31/01/2016	SNUBBING	5253860,54	5253860,54	2016	TARIF HORAIRE HRS 340 K	10080,00
22	2.2007.3	3	17150	31/01/2007	POMPAGE	7567288,89	7567288,89	2007	MOB. OU DEMOB. DE REGION A REGIOI	28600,00
23	2.2008.3	3	17150	31/01/2008	DIVERS	3351227,92	3351227,92	2008	MOB OU DEMOB (REGION A REGION)	22800,00
24	2.2009.3	3	17150	31/01/2009	TEXTEAM	7598784	7598784	2009	MOB. OU DEMOB. DE REGION A REGIOI	200,00
25	2.2010.3	3	17150	28/01/2010	TEXTEAM	4454287,2	4454287,2	2010	DTM DE PUIITS A PUIITS (AU DELA DE 5	121100,00
26	2.2011.3	3	17001	31/01/2011	SNUBBING	1751927,35	1751927,35	2011	MOBILISATION (MONTAGE ET TEST INC	1953,00
27	2.2012.3	3	17001	31/01/2012	SNUBBING	7198132,56	7198132,56	2012	DEMOBILISATION (DEMONTAGE ET CH/	10080,00
28	2.2013.3	3	17001	31/01/2013	POMPAGE	8087144,93	8087144,93	2013	OPERATING RATE / DAY (10 TO 12 JOUR	10080,00

Table 5. Business table, after join

After we proceed to join these tables to produce a global table of business that will be used for our experimentation.

Business table data can be incomplete, noisy, and inconsistent, which can affect the quality and efficiency of association rule retrieval. As a result, this table is

subjected to a set of data cleaning and purging operations.

Since the data of the business table is cleaned up, it is necessary to transform it into a format that can be used for the “Apriori” association rule extraction algorithm through a cross join.

The screenshot shows an Excel spreadsheet with the following data structure:

IDFACTU	FACT	Total de exist	BAC DE 50 M3 A MINIMUM 4H UN MOBILISATION H	UNITE SINGLE	PERS 2" X 15 K RELIEF VALV 2" X 15 K TEE	ADAPTER 4
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.1	SNUBBING	2				
2.2007.10	SNUBBING	1				
2.2007.10	SNUBBING	1				
2.2007.10	DIVERS	1				
2.2007.10	SNUBBING	1				
2.2007.10	SNUBBING	1				
2.2007.10	SNUBBING	1				
2.2007.10	SNUBBING	1				
2.2007.10	SNUBBING	1				
2.2007.100	SNUBBING	2				
2.2007.100	POMPAGE	2				
2.2007.100	POMPAGE	2				

Table 6. Business table with cross join

After, the data in this table must be converted into “.Arff” flat file format used by the Weka 3.8 tool.

The Business table in .Arff format has over 135 attributes like columns and over a 100 000 records as lines. In order to test the proposed approach, four samples of the business table were prepared, in order to compare the performance of

the developed system with a well-known classical algorithm for the extraction of association rules “Apriori”. The parameters taken to differentiate between the four samples are as follows: the seniority of data in the table, the number of records, the number of attributes and the size of the table. The characteristics of four business ERP data samples are presented from Table 7. to 10.

The screenshot shows the ARFF-Viewer interface with the following details:

- File: D:\DATA_Training\DATA_SNB_011.arff
- Relation: DATA_SNB_011-weka.filters.unsupervised.attribute.Remove-R1
- Attributes listed: 131: UNITE DE POMPAGE SUR SKID (HP), 132: UNITE DEPARAFFINAGE EN STAND BY, 133: UNITE DE POMPAGE EN STAND BY (AVEC PERSONNEL + VEHI), 134: ACT
- Attribute types: Nominal
- Values: t

Figure 16. Business table in .Arff format under Weka

Characteristics of business table 7	
Seniority of data	2 years
Number of attributes	135 attributes
Number of records	24 887 lines
Size of table	2 MB
Name of table	Business Table_7

Table 7. Characteristics of business table 1

Characteristics of business table 8	
Seniority of data	4 years
Number of attributes	135 attributes
Number of records	49 774 lines
Size of table	3.5 MB
Name of table	Business Table_8

Table 8. Characteristics of business table 2

Characteristics of business table 9	
Seniority of data	6 years
Number of attributes	135 attributes
Number of records	74 661 lines
Size of table	5.5 MB
Name of table	Business Table_9

Table 9. Characteristics of business table 3

Characteristics of business table 10	
Seniority of data	8 years
Number of attributes	135 attributes
Number of records	99 548 lines
Size of table	8.5 MB
Name of table	Business Table_10

Table 10. Characteristics of business table 4

4.4 Implementation of Distributed ERP Data Sites

The VMware® Workstation 12 Pro virtualization tool was used to create the distributed environment, in which the ERP data of the business table are partitioned and distributed on the different sites, to efficiently extract

business association rules from the ENSP business. In total, three data sites on three VMware virtual machines and one host data site on the host system are created in the VMware® Workstation environment. The configuration of the data sites is presented in Tables 11 to 14.

Data Site Description 1 (Host System)	
Hard disk	500Go
Store	16GB
Processors	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type of system	64 bits
Guest operating system	Windows 10 Professional

Table 11. Data Site Description 1 (Host System)

Data Site Description 2 (VMware)	
Hard disk	100 Go
Store	4GB
Processors	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type of system	64 bits
Guest operating system	Windows 8 Ultimate Edition

Table 12. Data site description 2

Data Site Description 3 (VMware)	
Hard disk	100 Go
Store	4GB
Processors	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type of system	64 bits
Guest operating system	Windows 8 Ultimate Edition

Table 13. Data site description 3

Data Site Description 4 (VMware)	
Hard disk	500 Go
Store	4GB
Processors	Intel(R) Core(TM) i7-3520M CPU @ 2.90 GHz
Type of system	64 bits
Guest operating system	Windows 8 Ultimate Edition

Table 14. Data site description 4

4.5 Interfaces of the Developed System

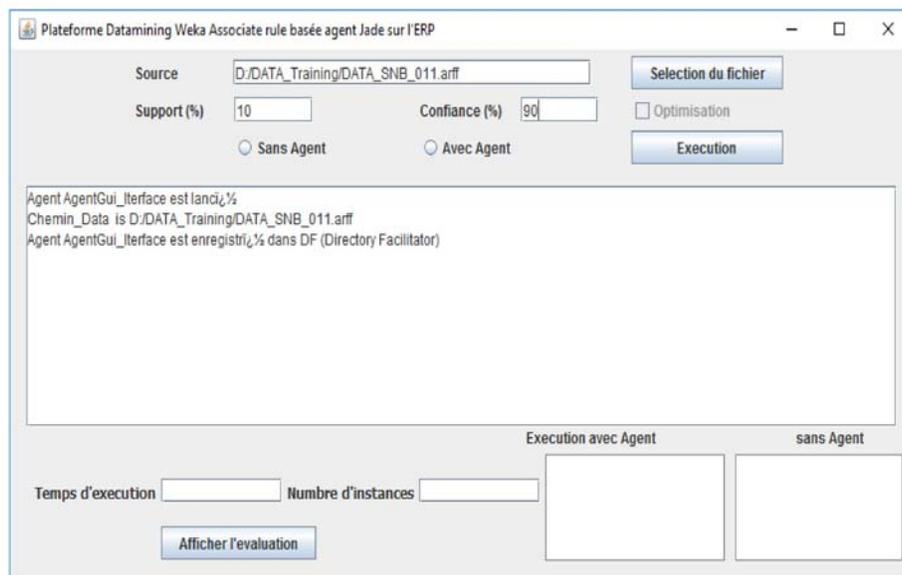


Figure 12. The main interface of the system developed

The following figure shows the first window that appears when we launch our system, which represents the interface of the DM Interface Agent and announces its start. In this window, the DM interface agent prompts the Data Miner user to enter the minimum support, the minimal confidence and the ERP data source in order to start the operation of extracting association rules by the business.

Since the parameters introduced, the DM user agent sends these parameters to the Data partitioning agent to start partitioning ERP data by business. When this agent finishes his work, he creates and launches, in a distributed and parallel manner, several association rule agents for each business partition. Each association rule agent begins extracting the business association rules, using the Apriori algorithm.

The association rule agents and the result integration agent communicate and cooperate throughout the process of association rule extraction. At the end of the operation, the DM interface agent displays the result of the extraction of business association rules, sent by the results integration agent.

4.6 Experiments

Our experiments carried out around four ERP data samples from the ENSP Business tables. The parameters taken into account in this analysis are the size of the ERP data samples in terms of a number of records, the size of the ERP data samples in terms of number of attributes and the age of data in the Business tables in terms of number

of years. All experiments performed on a host machine and three VMware virtual machines, running on Intel (R) Core (TM) i7-3520M CPU @ 2.90 GHz, 64-bit processor (s) and 16GB of RAM for the host system and 4GB for virtual machines, with Microsoft Windows 10 and 8 operating systems. All the ERP data samples from the described business tables distributed between the host system and the three virtual machines created. The following elements were measured in our experiments to show the performance of our system: (1) the response time (seconds/milliseconds), (2) the minimum support (3) the minimum confidence and (4) the number of records of the ERP data samples used.

4.7 Results and Discussions

In this section, we present the performance of our system by the results obtained from various experiments carried out on the ERP data of the business table. These experiments will be obtained by the variation of the minimum support, the minimum confidence and the number of records on the four business tables samples, namely the business tables 7, 8, 9 and 10.

The first experiment shows the results obtained by comparing the performance of the developed system with the classic “Apriori” algorithm. The minimum support threshold variation of 10% to 25% with the fixing of minimum confidence in 90%. Two samples of ERP data were used in this experiment such as: TableBusiness_7 and TableBusiness_8. (Figure 14. and Figure 15.)

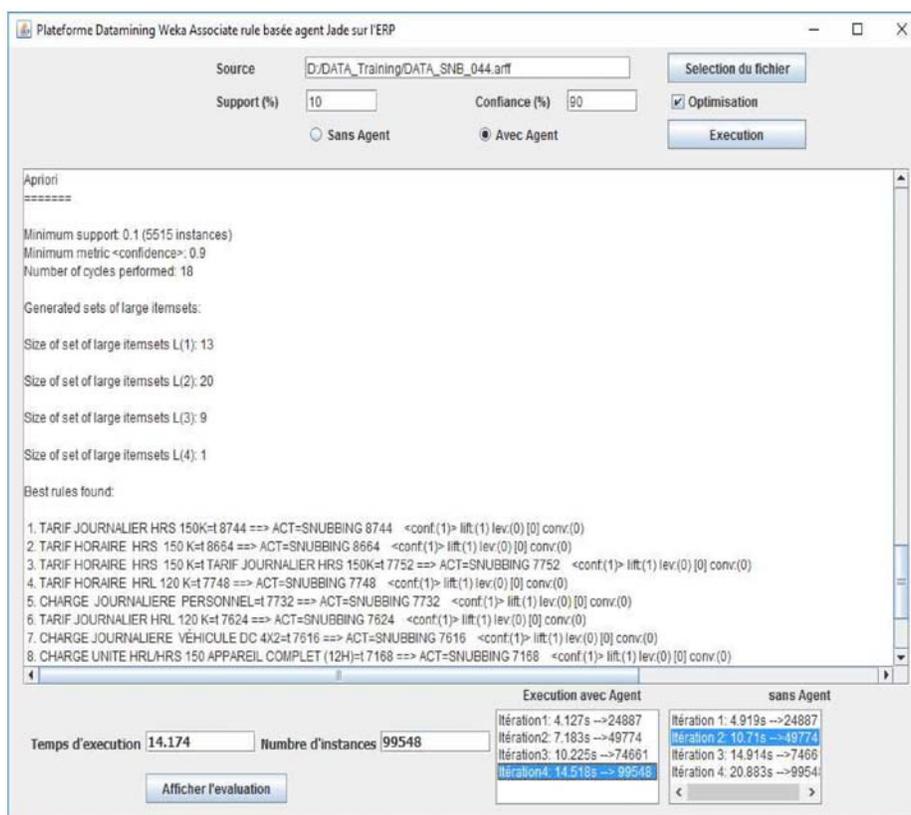


Figure 13. The main interface with the result of rule extraction

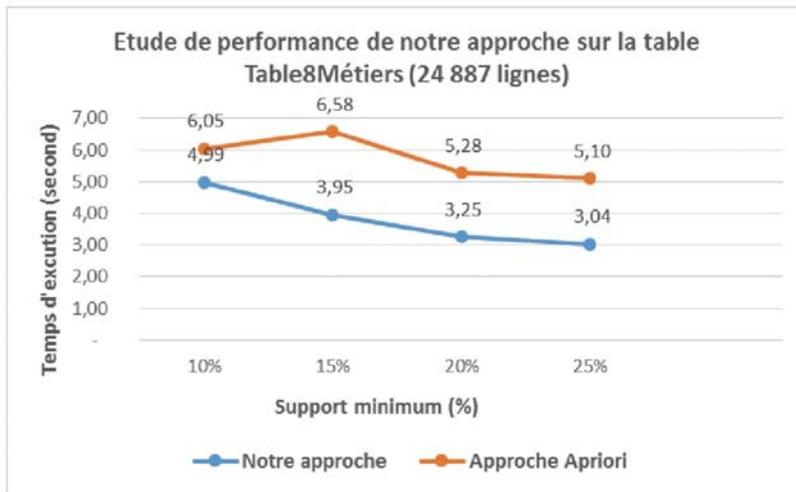


Figure 14. Performance study by varying the support of Table 7

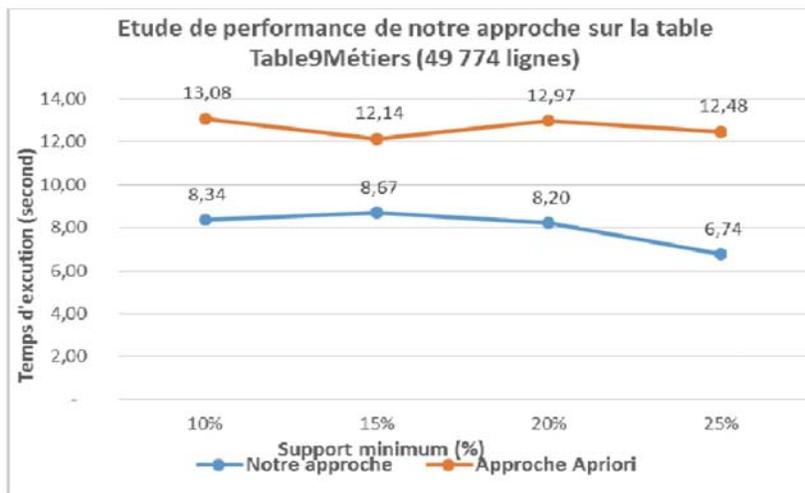


Figure 15. Performance study by varying the support of Table 8

The second experiment was also carried out on two samples of ERP data namely, the TableBusiness_10 and TableBusiness_11. This experiment is devoted to studying the performance of the proposed system by comparing it

with the “Apriori” algorithm, with the minimum confidence variation of 50% to 80% and the fixing of the minimum support to 10%. This experiment is shown in Figure 16 and 17.

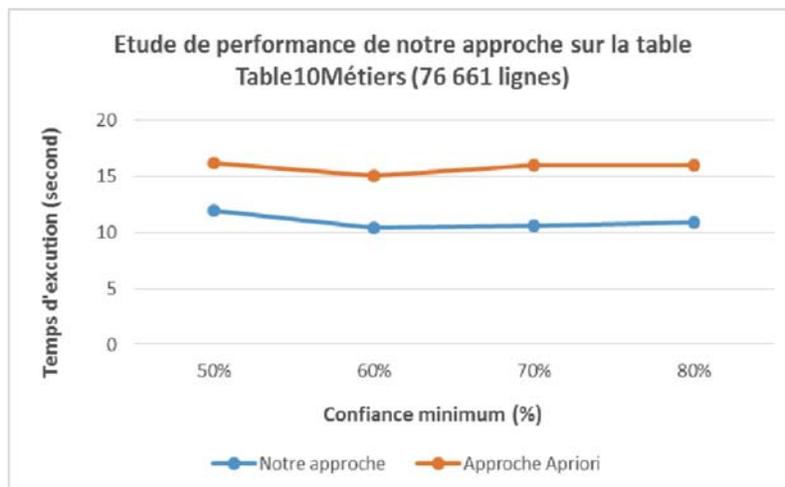


Figure 16. Confidence variation performance study in Table 9

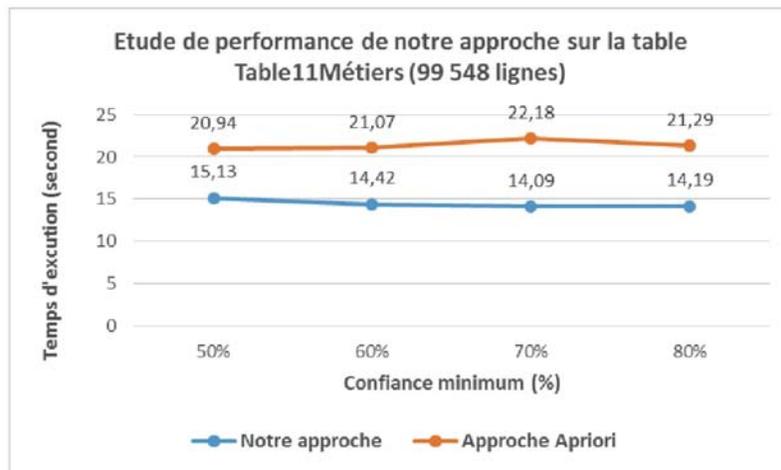


Figure 17. Confidence Variation Study in Table 10

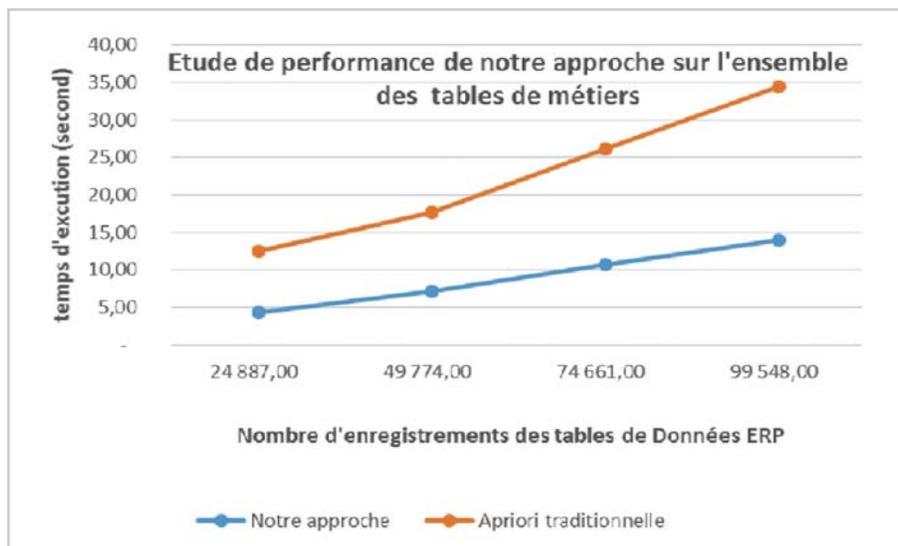


Figure 18. Performance study of our system on all samples of ERP data

Figure 18 shows the results obtained in the last experiment by the performance measurement of our system and compare it with the performance of the Apriori algorithm. This experiment is based on the change of a number of records on the four ERP data samples used, with fixing the minimum confidence setting at 90% and the minimum support at 10%.

The obtained results in the first two experiments show that the response time of our system is more efficient than the classic Apriori algorithm. The result of the last experiment has also shown that our system keeps its efficiency, even if the number of records increases over time, in the ERP database. Unlike the classic Apriori algorithm of knowledge extraction, that loses its effectiveness with the increase of the ERP database. The four experiments cited above show the effectiveness of our approach to optimize the business association rules extraction on the ERP database. This comes down to the use of multi-agent systems with the horizontal partitioning and the distribution of ERP data on separate machines.

4. Conclusion

Enterprises use different a huge amount of data in centralized databases systems in order to extract the relevant data this paper tackle this problem. In this work, we have presented an approach that uses agents paradigm with association rules to handle the problem of data relevant extraction. Moreover, this data extraction can help the managers and decision-makers to come up with a clear idea for good use of data. Thus, the degree of complexity it depends on data transactions in term of quantity it is getting larger either in the same sites or different sites of the company. The proposed architecture profits from agent properties such as flexibility, scalability, parallelism and intelligent interaction among different system's components. As we have seen in this paper using Apriori algorithm help in data relevant extraction especially in distribution after partitioning data into small parts to make the process easy than used to be. Furthermore, the agent paradigm is the best solution to simulate such kind of complex systems through the use

of its characteristics. In this paper, we proposed a new architecture based on multi-agents system, also this proposal applied on real case data from a very known petroleum company in Algeria at the national well services company (ENSP). The obtained results explain that our system is more efficient than the classic approach with Apriori algorithm in terms of accuracy. Installing the proposed system into practice could reduce the load during the preparation and delivery of business services within the ENSP company. Furthermore, it can serve as a decision support system for managers and engineers.

In the context of future work, as there are many petroleum companies in the Sahara of Algeria, we plan to integrate some other data analytics solution since we are working on data created daily. Also, we plan to propose a model for data streaming using deep learning method to analyze collected data in real-time to make the decision-making process quick, easier and accurate.

References

[1] Adhikari, A., Adhikari, J., Pedrycz, W. (2014). Data analysis and pattern recognition in multiple databases. *Intelligent systems reference library* 61 (Springer International Publishing, Switzerland, p 21–42.

[2] Agrawal, R., Shafer, J. (1996). Parallel mining of association rules. *In: Proceedings of the IEEE transactions on knowledge and data engineering*, 8(6) 962–9.

[3] Albashiri, K. A. (2010). EMADS an investigation into the issues of multiagent data mining. PhD thesis, the University of Liverpool, Liverpool L69 3BX, United Kingdom.

[4] Albashiri, K. A. (2014). Agent Based Data Distribution for Parallel Association Rule Mining. *In: International Journal of Computers*, 8.

[5] Albashiri, K. A. (2014). Agent based data distribution for parallel association rule mining. *Int J Compute*, 8 24-32.

[6] Albashiri, K. A., Coenen, F., Leng, P. (2010). An investigation into the issues of multi-agent data mining. *In: Agent based computing Nova Science Publishers*, 978-160876-684-0.

[7] Cheung, D. W., Xiao, Y. (1999). Effect of data skewness in parallel mining of association rules. *In: Proceedings of the 2nd Pacific-Asia conference on knowledge discovery and data mining*, Melbourne, Australia, p 48-60.

[8] Coenen, F., Leng, P. (2006). Optimising association rule algorithms using itemset ordering. *In: Proceedings of the AI Conference, Research and Development in Intelligent Systems XVIII. (Springer)*. p 53–66.

[9] Connell, S., Skillicorn, D. (2004). Building predictors from vertically distributed data. *In: Proceedings of the 2004 conference of the centre for advanced studies conference on collaborative research 04-07*. Markham, Ontario, Canada, p 150–62.

[10] Deypir, M., Sadreddini, M. H. (2009). Distributed

association rules mining using non derivable frequent patterns *Iranian Journal of Science and Technology, TransB: Eng* 33(6) 511-26.

[11] Ferber, J. (1999). *Multi-agent Systems - An Introduction to Distributed Artificial Intelligence*. Addison-Wesley, London.

[12] FERBER., J. (1995). *Les Systèmes Multi Agents : vers une intelligence collective*.

[13] Fiolet, V. (2006). Algorithmes distribués d'extraction de connaissances. Thèse de doctorat, Université de Mons-Mainault Lille1, France.

[14] Hamza, S., Abderaouf, G., Abdelhak, M., Okba, K. (2017, March). A new cloud computing approach based SVM for relevant data extraction. *In: Proceedings of the 2nd international Conference on Big Data, Cloud and Applications* (p. 1). ACM.

[15] Merizig, A., Kazar, O., Sanchez, M. L. (2019). A multi-agent system approach for service deployment in the cloud. *International Journal of Communication Networks and Distributed Systems*, 23(1) 69-92.

[16] Mesbahi, N., Kazar, O., Benharzallah, S., Zoubeidi, M., Rezki, D. (2018). A clustering approach based on cooperative agents to improve decision support in ERP. *Technological Innovations in Knowledge Management and Decision Support*, Book Chapter, IGI Publishing Hershey, PA.

[17] Mesbahi, N., Kazar, O., Benharzallah, S., Zoubeidi, M. (2015). A Cooperative Multi-Agent Approach-Based Clustering in Enterprise Resource Planning. *International Journal of Knowledge and Systems Science*, 6(1) 34-45.

[18] Mesbahi, N., Kazar, O., Benharzallah, B., Zoubeidi, M., Bouekkache, S. (2015). Multi-Agents approach for data mining based k-Means for improving the decision process in the ERP systems. *International Journal of Decision Support System Technology*, 7(2) 1-14.

[19] Ogunde, A., Folorunso, O., and Sodiya A. (2015). A partition enhanced mining algorithm for distributed Association rule mining systems. *Egyptian Informatics Journal* 16, 297–307.

[20] Ogunde, A., Folorunso, O., Sodiya A., Oguntuase, J., Ogunleye, G. (2011). Improved cost models for agent based association rule mining in distributed databases. *AnaleSeriInformatica* 9(1) 231–250.

[21] Pudi, V., Haritsa, J. (2003). ARMOR Association rule mining based on Oracle. *In: ICDM workshop on frequent itemset mining implementations*, Florida, USA.

[22] Symeonidis, A., Chatzidimitriou, C., Athanasiadis, I., Pericles, M. (2007). Data mining for agent reasoning: A synergy for training intelligent agents. *Engineering Applications of Artificial Intelligence (Elsevier)* 20(8)1097-1111.

[23] Yan, X., Zhang, C., Zhang, S. (2003). Towards databases mining: Pre-processing collected data. *Applied Artificial Intelligence*, 17 (5–6) 545–561.

- [24] YUAN.,T. (2008). Software Agents, Introduction to JADE.
- [25] Zaki, M. (2000). Parallel and distributed association mining: an introduction. *In: Proceedings of the large-scale parallel data mining*, lecture notes in artificial intelligence 1759. Berlin, Germany: Springer-Verlag, p 1-23.
- [26] Zhang, Z., Zhang, C., Zhang, S. (2003). An agent-based hybrid framework for database mining, *Journal of Applied Artificial Intelligence*, 17(5-6) 383–398.
- [27] Zoubeidi, M., Kazar, O., Benharzallah, S., Mesbahi, N., Merizig, A., Rezki, D. J. (2019). A new approach agent-based for distributing association rules by business to improve decision process in ERP systems. *International Journal of Information and Decision Sciences*.
- [28] Conrad, C., Ali, N., Kešelj, V., Gao, Q. (2016, August). ELM: An extended logic matching method on record linkage analysis of disparate databases for profiling data mining. *In: 2016 IEEE 18th Conference on Business Informatics (CBI)*, 1, 1-6 IEEE.
- [29] Pierrard, R., Poli, J. P., Hudelot, C. (2018, June). A Fuzzy Close Algorithm for Mining Fuzzy Association Rules. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems* (p. 88-99). Springer, Cham.