# DKDD_C: A Clustering- Based Approach for Distributed Knowledge Discovery

Marwa Bouraoui(✉), Houssem Bezzezi(✉), and Amel Grissa Touzi(✉)

Signal, Image and Technology of Information Laboratory, National Engineering School of Tunis, Tunis El Manar University, BP 37, Le Belvedere, 1002 Tunis, Tunisia
bourawimarwa@gmail.com, hbezzezi@gmail.com, grissa.touzi@topnet.tn

**Abstract.** In this paper, we address the problem of knowledge discovery. Several approaches have been proposed in this field. However, existing approaches generate a huge number of association rules that are difficult to exploit and assimilate. Moreover, they have not been proven themselves in a distributed context. As contribution, we propose, in this paper, DKDD_C, a new Distributed Knowledge Discovery approach. Exploiting, KDD based on data classification, we propose to give the choice to the user, either to generate Meta-Rules (rules between classes arising of preliminary data classification), or to generate classical Rules between distributed data. DKDD_C took place in both local and global processes. We prove that our solution minimizes the number of distributed generated association rules and then, offer a better interpretation of the data and optimization of the execution time. This approach has been validated by the implementation of a user-friendly platform as an extension of the Weka platform for the support of Distributed KDD.

**Keywords:** Distributed knowledge discovery · Mining association rules · Distributed database · Clustering · Weka plateform extension

## 1 Introduction

Nowadays, our ability to collect and store data from any type exceeds our possibilities of analysis, synthesis and Knowledge Discovery in Data (KDD). However, the performance of conventional centralized approaches degrade when the size of the processed data increases, in terms of execution time and memory space, hence we note the emergence towards the Distributed Knowledge Discovery (DKDD).

Several approaches and tools have been proposed in this context. Through our study, we found that these theoretical and practical approaches have different limits:

- Theoretically, DKDD algorithms generate a huge number of association rules that are difficult to exploit and assimilate.
- Practically, existing tools (1) support only some KDD algorithm that generates a large number of association rules that are difficult to assimilate (2) tools have not


Y. Tan et al. (Eds.): ICSI 2016, Part II, LNCS 9713, pp. 187–197, 2016.
DOI: 10.1007/978-3-319-41009-8_20

been proven themselves in a distributed context. (3) Are applied only to one restricted type of data.

We propose, in this paper, DKDD_C, a distributed knowledge discovery approach based on classification, which minimizes the number of distributed generated association rules and then offer a better interpretation of the data and optimized both the space memory and the execution time. By exploiting, KDD based on data classification, we propose to give the choice to the user, either to generate Meta-Rules (rules between classes arising of preliminary data classification), or to generate Rules between distributed data without preliminary classification. This approach has been validated by the implementation of a user-friendly plat-form as an extension of the Weka platform for the support of DKDD.

This paper is organized as follows: Section 2 presents some basic concepts for the DKDD, Sect. 3 presents our motivation, Sect. 4 provides a description of our proposed approach, and Sect. 5 presents the implementation and validation of our approach. We end with a conclusion and some perspectives.

## 2 Basic Concepts

In this section we present the basic concepts related to our research.

### 2.1 Mining Distributed Association Rules

Association Rules Mining is an important problem in the field of knowledge discovery aims at finding meaningful relationships between the attributes of databases in order to identify the groups of items that are most frequently purchased together. The first effective algorithm is Apriori [5]. Other algorithms have been proposed to improve performances such as CHARM [9] and Closet [10].

However, given the increase of the size of the processed data which lengthen the execution time and fill the memory space, the performances of centralized conventional approaches are deteriorating increasingly. Thus, new algorithms are opting for the parallelization and the distribution of this research problem.

Several approaches have been proposed in the literature. The first proposed algorithm is CD [6], which present a simple distribution of Apriori [5] algorithm. Other algorithms have been proposed later to propose more effective solutions. These algorithms include FDM [8] which introduce powerful pruning techniques called global and local pruning that minimize the size of candidates. ODAM [11] which reduce average size of transactions by eliminating all infrequent items from DB transactions after the first pass and therefore efficiently generate candidate support counts in latter passes. EDFIM [2] extended the ODAM algorithm to all passes and then offer better results. We can cite also the L-Matrix algorithm [7] that reduces the time of scan of database partition by using LMatrix to find support counts instead of scanning the databases partition time after time, which will save a lot of memory. In [3], DDRM algorithm is proposed that is a dynamic extension of Prefix-based [4] algorithm and has used a lattice-theoretic approach for mining association rules.

### 2.2 Clustering

Clustering is an important task in data mining that aims to organize data into groups of similar observations. Clustering approaches can be classified into two categories, a crisp approaches that consist to associate each object to a single cluster (such as K-means [13]), and fuzzy clustering (such as Fuzzy K-means [12]), that deals with overlapping data clusters. The main advantage of this type of data clustering is that it gives the flexibility to express that data points can belong to more than one cluster.

### 2.3 Panorama of Existing Platforms

The growing interest of the Data Mining method was accompanied with the appearance of many tools. Among them, we can mention:

- WEKA: This tool includes a set of training algorithms of Data Mining covering some supervised and not supervised methods of classification.
- TANAGRA: Free Data Mining software for academic and research purposes. It proposes several data mining methods for exploratory data analysis, statistical learning, machine learning and databases area.
- ORANGE: Open source data visualization and analysis for novice and experts. Data mining through visual programming or Python scripting. Components for machine learning.

Through our study, we see a height convergence to the Weka tool, and this is for several reasons including: (1) the diversity of its implemented data mining algorithms compared to other tools (2) it implements two type of clustering unlike other tools that don't implement the fuzzy clustering (3) it is very portable as it is completely implemented in Java and therefore runs on all modern platforms (4) it is a free tool.

## 3 Motivation

Several approaches and tools have been proposed for the DKDD. Through our study, we found that these approaches have different limits:

- Theoretically, DKDD algorithms generate a huge number of association rules, which causes (1) this requires a large memory space for data modeling, and data structures required by these algorithms such as trees or graphs or trellis. (2) The execution time for the management of these data structures is important. (3) Generated rules from these data are generally redundant rules. (4) Algorithms of mining of association rules give a very large number of rules that are difficult to assimilate.
- Practically, existing tools (1) support only some KDD algorithm that generates a large number of association rules that are difficult to use (2) tools have not been proven themselves to a distributed context. (3) Are applied only to one restricted type of data (4) they produce the output on text mode, so there are no the visualization of association rules (5) do not provide the calculation of the execution time.

We propose, in this paper, DKDD_C, a distributed knowledge discovery approach based on classification, which minimizes the number of distributed generated association rules and then offer a better interpretation of the data and optimized both the space memory and the execution time.

This approach has been implemented under the centralized data mining tool Weka, that we have extended it to support DKDD. The user interface is a modified Weka Explorer environment that supports not only the execution of both local and global data mining tasks, but also present solutions for practical existing tools limits cited above, such as graphic visualization of association rules, calculation of execution time, loading data from various source and different type of data.

## 4 New Approach

In this section, we present DKDD_C, our new approach of Distributed KDD.

### 4.1 General Principe

Our approach provide a solution for Distributed KDD and more precisely an effective generation of distributed association rules. The DKDD_C process takes place in two main phases:

- **Data collection phase** which consist in the preparation and collection of necessary information for the distributed knowledge extraction phase.
- **Distributed knowledge extraction phase**, which is divided into two distinct phases according to the user's choice:
  - Local DKDD phase: for knowledge discovery from a specific site of the DDB.
  - Global DKDD phase: for knowledge discovery from all sites of the DDB.

In the knowledge discovery phase, we added a classification step as a preliminary step to the mining of association rules. By exploiting, KDD based on data classification, we propose to give the choice to the user, either to generate Meta-Rules (rules between classes arising of preliminary data classification), or to generate Rules between distributed data without preliminary classification.

The Meta-Knowledge concept (Meta-Rules) models a certain abstraction of the data that is fundamental when the number of data is enormous. Moreover, the number of generated rules is smaller with this concept. Indeed, theoretically, while classifying data, we construct homogeneous groups of data. These groups, called clusters, have the same properties, so defining rules between these clusters implies that all the data elements belonging to those clusters will be necessarily dependent on these same rules. Thus the number of generated rules is smaller since one processes the extraction of the knowledge on the clusters which number is relatively lower compared to the initial data elements.

### 4.2 Architecture of the New Approach

In this section, we present the general architecture of our approach as shown in Fig. 1.

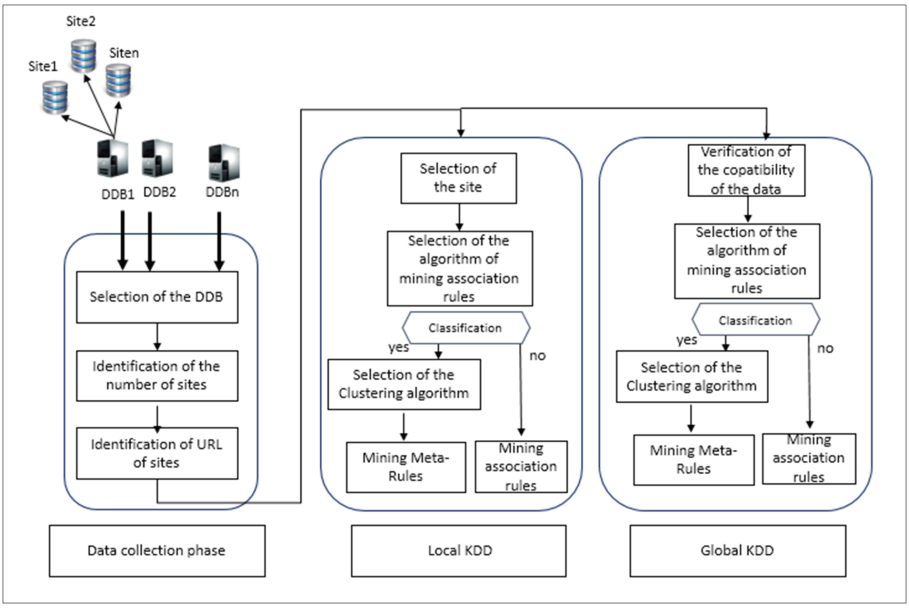


**Fig. 1.** General architecture of DKDD_C

**Data collection phase.** The data collection phase is the first phase of our knowledge discovery process. It consist in the preparation and collection of necessary information for the distributed knowledge discovery phase. It takes place in three steps as follows:

**Step 1:** The selection of the DDB on which we wish to apply the DKDD process.
**Step 2:** The identification of the site number of the DDB to define the data location in each remote site.
**Step 3**: The identification of the URL of each database to enable the remote access to databases of remote sites and import data on which we wish extract knowledge.

**Distributed Knowledge Extraction phase.** This phase is the main phase of our approach that is divided into two distinct processes according to the user's choice: Local and Global DKDD process. We detail in the following sections these two processes.

*Local KDD process.* This phase handles extracting knowledge from distributed data from a specific site selected by the user. It takes place in five steps, as follows:

**Step 1**: The selection of the site on which we want to apply the KDD process.
**Step 2**: The selection of the algorithm of association rules mining the most optimal according to the field of data.
**Step 3**: The selection of the classification algorithm
**Step 4:** Classification of the data of selected site.
**Step 5**: Local KDD: Applying the selected algorithm of association rules mining for the selected site in Step 1, and so generate Meta-Rules (rules between clusters obtained in Step 4) or classical Rules if the user choose to neglect the Step 4.

*Global KDD process.* This phase handles knowledge discovery from all sites of the distributed database selected by the user. It takes place in five steps, as follows:

**For each site:**
**Step 1:** Checking of the compatibility of distributed data: it consist in checking the consistency of distributed data in different sites: (1) Check the type of fragmentation of the DDB. (2) Check the various entities (Tables / Attributes) in the DB of each site.
**Step 2**: The selection of the algorithm of association rules mining the most optimal according to the field of data.
**Step 3**: The selection of the classification algorithm
**Step 4:** Classification of the data of each site
**End For**
**Step 5:** Global KDD
**For each site:** Applying the selected association rules mining algorithm and so generate Meta-Rules or classical Rules if the user choose to neglect the Step 4.
**End For**

The classification step in the knowledge extraction phase, Step 3 for both Local and Global process, is depending on the user' choice. This classification phase is proposed in Grissa [1] for centralized databases. We then have added this model as a step in our Distributed KDD approach (Fig. 2).

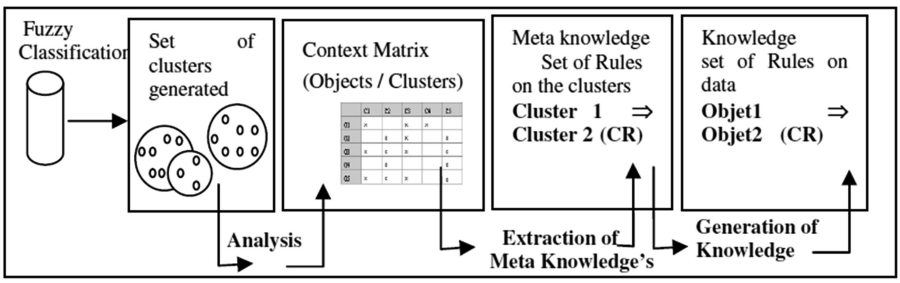


**Fig. 2.** The KDD approach based on classification

The different steps of this model are conducted as follows:

**Step 1:** Enter the datasets (any type of data).
**Step 2:** Apply a fuzzy clustering algorithm to organize data into groups (or clusters).
**Step 3**: Determine the fuzzy formal context of the matrix obtained in step 2.
**Step 4:** Apply on the matrix obtained an algorithm of association rule generating for the KDD process in the form of association rules between clusters.
**Step 5**: Generate knowledge of the dataset as association rules.

## 5 Implementation and Validation of the New Approach

### 5.1 Implementation.

For the implementation of our approach, we have worked with the Weka platform Version 3.6.12. We detail below the functioning of our solution:

**Data collection and preparation phase.** For managing distributed data, we have extended the original interface of Weka in which we added the option "Distributed Data" (Fig. 3). The user can consult existing DDB information: the number of distribution sites and their URLs (Fig. 4).

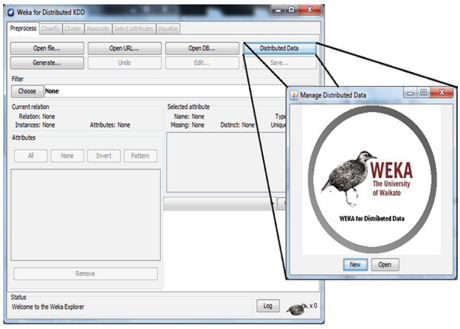


**Fig. 3.** Access to distributed process

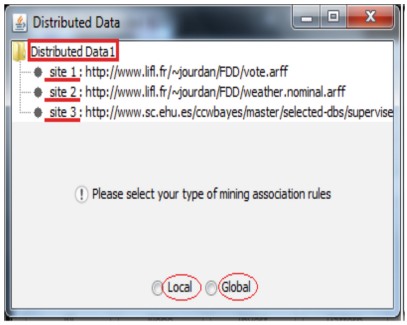


**Fig. 4.** Selecting the DDB

Our solution works in two main phases, namely Local and Global DKDD and this according to the user needs (Fig. 4).

**Local DKDD.** This phase consist in managing the distributed data by site, and this in order to visualize the result of data mining of each site separately. By selecting a specific site, our solution imports its data through its URL. Figure 5 shows the loading of data of site 2 in the Weka data preprocessing panel.

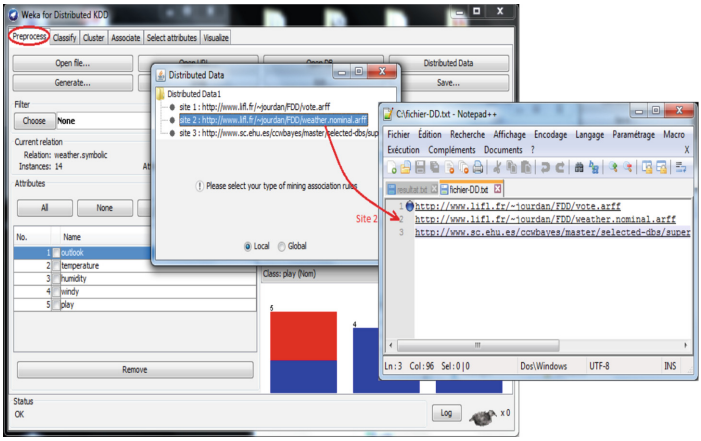


**Fig. 5.** Loading of sites of the selected DDB

Thus the user selects the classification algorithm and that of association rule mining (algorithms already implemented in Weka) that we have operated for the generation of Meta-Rules (Fig. 6). When achieving Local DKDD process, Association Meta-Rules are displayed in the "Out Put" panel of Weka (Fig. 7)

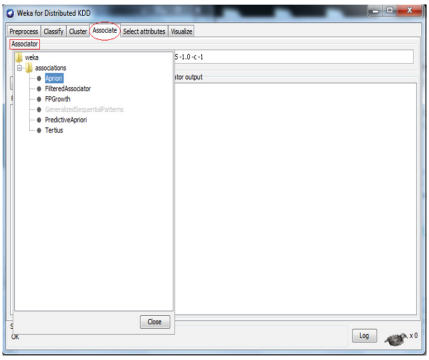


**Fig. 6.** Selection of rules mining algorithm

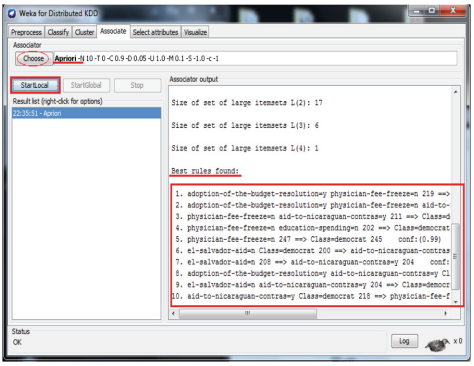


**Fig. 7.** Local mining of association rules

**Global DKDD.** This is the principal phase of our approach, our solution supports distributed data retrieval site by site just by giving the user the possibility to confirm the import of data from sites. Figures 8, 9 and 10.

By specifying the algorithms of classification and association rules extraction, our solution performs the classification process according to the selected algorithm, then generates the Meta-Rules between clusters obtained according to the selected algorithm of association rules mining (Fig. 11).


**Fig. 8.** Loading of sites

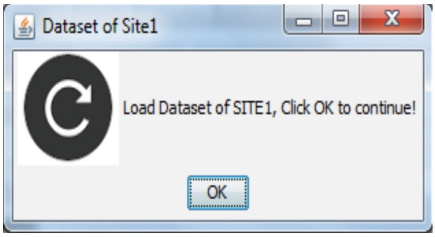


**Fig. 9.** Loading of data of site1

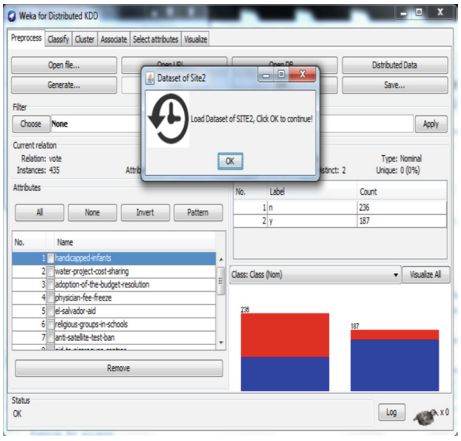


**Fig. 10.** Global execution

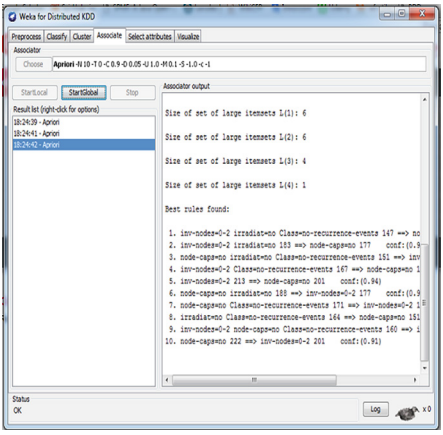


**Fig. 11.** Global execution

## 5.2 Validation

The proposed approach presents a number of advantages which can be summarized as follows:

- **The concept of "Global" and "Local" KDD:** The concept of "Global" KDD enables a global view of the DDB dataset and the concept of "Local" KDD allows a detailed view of initial database dataset on each remote site.
- **The definition of KDD approach based on classification (definition of Meta-Knowledge concept):** The Meta-Knowledge concept is very important in the case of very voluminous dataset because it gives a global view of this data set and models a certain abstraction of the data that is fundamental when the number of data is enormous. Moreover, the number of generated rules is smaller with this concept.
- **The extensibility of this approach:** In our approach, the step of generating association rules can be applied with any KDD algorithm. Studies have shown that, the fact that an algorithm is more optimum than another is according to the field of

used data. This means that we can apply the most optimum method according to the field of the dataset. In addition, the data classification step in our KDD process can be applied with any fuzzy clustering algorithm to classify the initial data.

The performance of our approach was evaluated on two datasets, "Car" which is derived from the simple hierarchical decisional model, and "Mushrooms" which is a set of dense data on fungi describing their characteristics. We have installed these datasets at each site of the DDB (site number = 3). As classification algorithm, we have tested with FCM (fuzziness = 2 and accuracy = 0.002). As algorithm of association rules extraction, we have tested with Apriori and Close algorithms. The parameters for these two algorithms are, confidence = 10 % and support = 0.5 % with "Car" datasets and confidence = 70 % or 50 % and support = 10 % with "Mushrooms" datasets.

To test performances according to the number of generated association rules (NAR) we have used both "Car" and "Mushrooms" datasets. We have tested our approach with different number of cluster (NC), we show here tests with 10 and 15 clusters (Fig. 13). According to the tests we carried out, we can conclude that the number of rules generated while applying our DKDD_C approach is reduced relative to the number of rules generated in the case of a direct application of an association rule mining algorithm. This reduction of the number of rules is due by the intervention of the classification step (fuzzy clustering in our tests). Indeed, the number of generated rules is smaller since one processes the extraction of the knowledge on the clusters which number is relatively lower compared to the initial data elements.

For execution time test, we have used "Car" datasets. Tests show that the execution times of these algorithms are increasing in function of the number of selected clusters, and they are lower than the execution times generated by these algorithms without the clustering step (using conventional approaches) for a reduced number of clusters (Fig. 12). This reduction of the execution time is due by the minimizing of the number of generated rules.

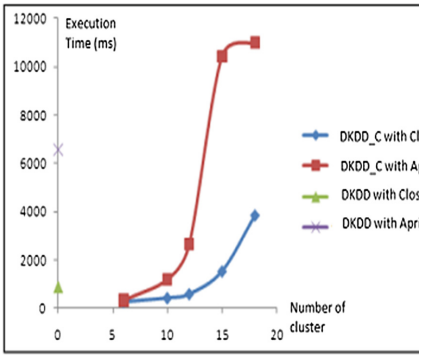


**Fig. 12.** Execution time

| Datasets MinSupp | MinConf | Apriori | | | Close | | |
|---|---|---|---|---|---|---|---|
| | | DKDD | DKDD_C | | DKDD | DKDD_C | |
| | | NAR | NC | NAR | NAR | NC | NAR |
| Muchroom 10% | 70% | >110000 | 10 | 1185 | 48401 | 10 | 371 |
| | | | 15 | 12711 | | 15 | 3144 |
| | 50% | >110000 | 10 | 1808 | 92651 | 10 | 581 |
| | | | 15 | 16533 | | 15 | 4212 |
| Car 0.5% | 10% | 42315 | 10 | 3794 | 35234 | 10 | 139 |
| | | | 15 | 23128 | | 15 | 3753 |

**Fig. 13.** Number of generated rules

## 6 Conclusion

We proposed, in this paper, DKDD_C, a distributed knowledge discovery approach based on classification, which minimizes the number of distributed generated association rules. Indeed, based on, KDD based on data classification, we propose to give the choice to the user, either to generate meta-rules, that is to say, generate rules between classes arising of preliminary classification on distributed data (Meta-Rules), or to generate rules between data distributed without preliminary classification. This approach has been implemented under the centralized data mining tool Weka, that we have extended it to support DKDD from different type of distributed data.

As perspective, we propose (1) to extend our approach to dynamic databases with a variable number of sites (2) to test and adapt our approach for large distributed databases.

## References

1. Grissa, A.: Towards a discovering knowledge comprehensible and exploitable by the end-user. In: 2nd International Conference on Advances in Databases, Knowledge, and Data Applications, pp. 126–134 (2010)
2. Adelpoor, A., Abadeh, M.S.: An efficient frequent itemsets mining algorithm for distributed databases. Int. J. Comput. Sci. Electron. Eng. (IJCSEE) **1**(1) (2013)
3. Wessel, T.: Parallel mining of association rules using a lattice based approach. Nova Southeastern University (2009)
4. Zaki, M.J.: Hierarchical parallel algorithms for association mining. In: Kargupta, H., Chan, P. (eds.) Advances in Distributed and Parallel Knowledge Discovery, pp. 336–339. MIT Press, Cambridge, MA (2000)
5. Agrawal, R., Skirant, R.: Fast algorithms for mining association rules. In: Proceedings of the 20th International Conference on Very Large Databases, pp. 478–499 (1994)
6. Agrawal, R., Shafer, J.C.: Parallel mining of association rules. IEEE Trans. Knowl. Data Eng. **8**(6), 962–969 (1996)
7. Arokia Renjit, J., Shunmuganathan, K.L.: Mining the data from distributed database using an improved mining algorithm. Int. J. Comput. Sci. Inf. Secur. (IJCSIS) **7**(3), 116–121 (2010)
8. Cheung, D.W., Han, J., Ng, V.T., Fu, A.W., Fu, Y.: A fast distributed algorithm for mining association rules. In: Proceedings of the Fourth International Conference on Parallel and Distributed Information Systems, pp. 31–42 (1996)
9. Zaki, M.J., Hsiao, C.J.: CHARM: An efficient algorithm for closed itemset mining. In: The 2nd International Conference on Data Mining, Arlington, pp. 34–43 (2002)
10. Pei, J., Han, J., Mao, R., Nishio, S., Tang, S., Yang, D.: CLOSET: an efficient algorithm for mining frequent closed itemsets. In: Proceedings of the ACM SIGMOD DMKD 2000, Dallas, TX, pp. 21–30 (2002)
11. Ashrafi, M.Z., Taniar, D., Smith, K.: ODAM: an optimized distributed association rule mining algorithm. IEEE Comput. Soc. **5**(3), 1541–4922 (2004). IEEE Distributed Systems
12. Bezdek, J.: Fuzzy mathematics in pattern classification. Ph.D. Dissertation, Cornell University (1973)
13. McMueen, J.: Some methods for classification and analysis of multivariate observations. In: The Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297 (1967)