

www.arpnjournals.com

## A STUDY ON DISTRIBUTED DATA MINING FRAMEWORKS

Srimathi C.<sup>1</sup>, M. Subaji<sup>2</sup>, Anu Soosan Baby<sup>1</sup> and Deepu Raveendran<sup>1</sup> <sup>1</sup>School of Computer Science and Engineering VIT University, Vellore, India <sup>2</sup>International Relation, VIT University, Vellore, India E-Mail: <u>csrimathi@vit.ac.in</u>

### ABSTRACT

Distributed Data Mining is an interesting research community with respect to next generation of computing platform such as SOA, Grid and Cloud etc. and important paradigms such as Peer to Peer. There are mainly three types of Distributed Data Mining algorithms: DDM based parallel data mining agent, DDM based on mete learning, DDM based on Grid. In this paper analyses the key issues in Distributed Data Mining and a detailed survey on three kinds of DDM algorithms, its performance and its various pros and cons. The paper finally comes with an essential characteristics needed for a good DDM algorithm.

Keywords: distributed data mining, MADM, CAKE, JAM, VEGA.

#### **1. INTRODUCTION**

Distributed Data Mining is a field of data mining that offers a framework to mine distributed data paying careful attention to distributed data and computing resources. Distributed data mining is used to extract patterns and regularities from large number of distributed data sets (Bin Liu *et al.*, 2010). Traditional algorithm that works in centralized environment is proved to be unsuitable for mining in distributed settings. Distributed Data Mining algorithm focuses on analysis and modeling of distributed data (Dam H.H. *et al.*, 2005), (Prodromidis A. L. *et al.*, 2000).

Most of the algorithms designed to operate on distributed databases by downloading the relevant data to a centralized location and then perform data mining. But these algorithms create many issues regarding bandwidth and privacy protection, since a sufficient bandwidth is required to transfer huge size of data and unable to preserve privacy of sensitive data. Analyzing distributed data is a nontrivial problem (Prodromidis A. L. *et al.*, 2000).

In DDM, one of the two assumptions is commonly adopted as how data is distributed across sites: (horizontally homogeneously partitioned) and heterogeneously (vertically partitioned). Both adopt a conceptual view point that the data tables at each site are partition of a single global table. In homogeneous case, global table is horizontally partitioned (Dam H.H. et al., 2005). The tables at each site are subsets of a global table; they have exactly the same attributes. In heterogeneous case Table is vertically partitioned, each site contains a collection of columns. However each tuples at each site is assumed to contain a unique identifier to facilitate matching (Prodromidis A. L. et al., 2000).

Some features of distributed scenario where Distributed Data Mining are applicable are: A system consists of multiple independent sites of data and computation which communicate only through message passing, Communication between sites is expensive, and Sites have resource and privacy concerns. The primary goal of many distributed data mining method is to reduce the number of messages send. Some methods attempt to load balance across the sites to prevent performance from being dominated by time and space usage of any individual sites (Giannella C. *et al.*, 2004).

There is a need for Distributed Data Mining because centralization is hard as it requires multi terabytes data to be transmitted over long distances. Centralization violates privacy legislation, exposes business secrets or poses other social challenges. A need exists for developing algorithm tools services and infrastructure that let us mine data distributed across organizations (Prodromidis A. L. *et al.*, 2000), (Giannella C. *et al.*, 2004).

#### 2. STEPS IN DISTRIBUTED DATA MINING

Data preprocessing is an important step in Data Mining. Since data in the distributed environment is noisy, incomplete, inconsistent we need data preprocessing. Major tasks in data preprocessing involves Data Cleaning, Data Integration, Data Transformation, Data Reduction and Data Discretization (Giannella C. *et al.*, 2004).

**Step-1:** Data Cleaning is the basis step for any Data Mining algorithm. It involves fill in missing values, identify outliers and smooth out noisy data, correct inconsistent data, resolve redundancy caused by data integration. Data Cleaning is performed at each local site in a distributed algorithm (Bin Liu et al., 2010).

**Step-2:** Data Clustering is an unsupervised classification. Data Clustering is used to place data elements into related groups without advance knowledge of group definitions. In Distributed Data Mining common data clustering used is K- means algorithm (Bin Liu et al., 2010).

**Step-3:** Data Integration combines data from multiple source into a coherent store. It integrates metadata from different sources. In distributed scenario, data may be heterogeneous and data integration requires combining data from multiple sources. In distributed scenario, integration may results in redundancy of multiple databases (Bin Liu *et al.*, 2010).



#### www.arpnjournals.com

**Step-4:** After data combined from multiple sources Data Transformation is performed. Data Transformation involves Smoothing, Aggregation, Generalization and Normalization (Bin Liu et al., 2010).

**Step-5:** In Data reduction a reduced representation of data set which is smaller in volume but yet produces same result. This is essential in Distributed Data Mining since it creates huge amount (Tera-bytes) of data, and it is highly necessary to convert to a reduced representation (Bin Liu et al., 2010).

**Step-6:** Data Discretization is essential before preprocessing for further analysis. In data Discretization, we use some classification algorithm to reduce the data size by Discretization (Bin Liu et al., 2010).

#### 3. KEY ISSUES IN DISTRIBUTED DATA MINING

The major issues that affect the performance of Distributed Data Mining are follows:

**Heterogeneous data mining:** If the data is heterogeneous there will be contradiction among the attributes will occur. If the data is heterogeneous local data management model should be integrated into a global model before dealing with the data (Prodromidis A. L. *et al.*, 2000).

**Data consistency:** Since data is distributed across many sites, it creates a problem of data inconsistency. If the modification done to the local data model cannot be notified to global data model it may affects the final output produced after Data Mining (Prodromidis A. L. et al., 2000).

**Communication cost:** Communication cost depends on network bandwidth and amount of transferred information. In Distributed Data Mining a cost model should be built (Prodromidis A. L. et al., 2000).

**Knowledge integration:** In knowledge Integration the local results are integrated to produce global results. It is the critical step in any Distributed Data Mining. During integration the local sample should not lose its value in the global range, it must be preserved (Giannella C. *et al.*, 2004).

**Data variance:** In distributed environment data is not static as that of traditional data mining. Along with data the executing environment is also dynamic; hence the Distributed Data Mining algorithm should correctly transfer the time series result and time series related result (Giannella C. et al., 2004).

**Privacy preserving:** The main objective of privacy preserving data mining is to develop algorithm for modifying original data in some way so that private data and private knowledge remain private even after mining process. The problem that arises when confidential information can be derived from released data by unauthorized users is commonly called database inference problem. Most recent efforts addressing the privacy issue involves sanitation, data distortion and cryptographic methods (Giannella C. *et al.*, 2004).

### 4. CLASSIFICATION OF DISTRIBUTED DATA MINING ALGORITHMS

Before the distributed scenario, centralization exists. Centralization is hard since it requires multi tera byte data set to be transmitted over very long distances. Centralization violates privacy legislation, exposes business secrets or poses other social challenges. A need exists for developing algorithmic tools services and infrastructure that let us mine data distributed across organizations (Dam H.H. *et al.*, 2005).

**Distributed data mining algorithm is classified into three classes:** DDM based on Multi Agent System, DDM based on Meta learning, DDM based on grid (Bin Liu *et al.*, 2010).

# 4.1. DDM based on multi agent system: MADM and CAKE architectures

Multi Agent System (MAS) offer architecture for collaborative problem solving in distributed environments. The behavior of agents depends on data from distributed sources. Agents in MAS need to be pro-active and autonomous. Agents perceive their environment, dynamically reason out actions based on conditions and interact with each other. In some applications performance of agents depends on existing domain theory (Prodromidis A. L. *et al.*, 2000).

Knowledge of MAS is complex and collective, complex in the sense that it is the outcome of data analysis and preexisting domain knowledge. Analysis of data may require advanced data mining for detecting hidden patterns, constructing predictive models and identifying outliers among others. Collective means each analysis is performed by different agents (Prodromidis A. L. et al., 2000), (Giannella C. et al., 2004). MAS are mostly used in sensor nodes where there is a requirement for comparison of data at different nodes. Since solution for distributed problem requires collaboration, semiautonomous behavior and reasoning, there is a perfect synergy existing between MAS and DDM (Prodromidis A. L. et al., 2000). Agents are used in this system for the following purposes.

Autonomy of data sources: A Distributed Mining (DM) agent may handle access to underlying data source in accordance with given constraints on required autonomy of system, data and model.

**Interactive DDM:** Interactive DDM allows a human user to supervise and interfere with running data mining process (Giannella C. *et al.*, 2004).

**Dynamic election of source and data gathering:** DM agents applied to adaptively select data sources according to given criteria such as expected amount, type and quality of data at considered source, actual network and DM server load (Giannella C. *et al.*, 2004).



#### www.arpnjournals.com

Scalability of DDM to massively distributed data: A set of DM agents allow for a divide and conquer approach by performing mining task locally to each of data sites. DM agents aggregate relevant pre-selected data to their originating server for further processing and may evaluate best strategy between working remotely or migrating on data sources (Giannella C. *et al.*, 2004).

**Multi s trategy DDM:** Data Mining agents uses different data mining technique to choose depending on type of data retrieved from different sites. The learning of MAS is based on multi strategy selection of DM methods (Giannella C. *et al.*, 2004).

**Security:** Agent Code and data integrity is a crucial issue in secure DDM. Subverting or hijacking a data mining agent places a trusted piece of software. If DM agents are even allowed to migrate to a remote computing environments methods to ensure authentication and confidentiality of mobile agents have to be applied. Selective agent replication may prevent malicious host from blocking or destroying temporarily residing DM agents (Giannella C. *et al.*, 2004).

**Trustworthiness:** DM agents may infer sensitive information even from partial integration to a certain extent, with some probability. It is called inference problem. It enables us to integrate implicit knowledge from different source using commonly held rules of thumb (Giannella C. *et al.*, 2004).

DDM applications can be enhanced with agents. Agent based DDM takes data mining as basis foundation and inherits all powerful properties of agents. Constructing an ADDM system concerns three key characteristics: Interoperability, Dynamic System Configuration and Performance aspects (Giannella C. *et al.*, 2004). Interoperability concerns, not only collaboration of agents in the system, but also external interaction which allow new agents to enter the system seamlessly. The architecture of system must be open and flexible so that it can support interaction including communication protocol, integration policy and service directory (Giannella C. *et al.*, 2004).

Dynamic System Configuration, tend to handle dynamic configuration of system, is a challenge issue due to complexity of planning and mining algorithm. A mining task may involve several agents and data sources, in which agents are configured to equip with an algorithm and deal with given data set (Giannella C. *et al.*, 2004) .Performance, can either be improved or impaired because the distribution of data is a major constraint. In distributed environment, task can be executed in parallel, in exchange concurrency issues arise. Quality of Service Control in performance of data mining and System Perspectives is desired.



Figure-1. Overview of Agent based Distributed Data Mining System (Prodromidis A. L. *et al.*, 2000).

**Data:** In Distributed Environment, data can be hosted in various forms such as online relational databases, data stream, web pages etc.

**Communication:**In distributed environment, communication protocol may vary depending on the implementation of system, such as client-server, peer-to-peer.

**Presentation:** Interface simplifies complex distributed systems into user friendly message such as network diagrams, visual reporting tools etc.

**Query optimization:** Query Optimization determines the type of mining tasks and chooses proper resources according to the request. It also determines whether it is possible to parallelize the tasks, since the data is distributed and can be mined in parallel.

**Discovery plan:** It allocates sub tasks with related resources. Since sub tasks as well as their results are performed asynchronously mediating agents play important role in coordination.

**Local knowledge discovery:** At each data site, mining process may take place locally depending on the individual implementation, and these patterns are transferred over the network.

**Knowledge discovery:** It is also known as data mining, it executes the algorithm as required by task to obtain knowledge from specified data source.

**Knowledge consolidation:** It is necessary to normalize the knowledge obtained from various sources in order to get meaningful mining result. It involves a complex methodology to combine knowledge or patterns from distributed sites.

#### www.arpnjournals.com



Figure-2. MADM systems architecture (Giannella C. *et al.*, 2004).

**Interface Agent:** It interacts with user (or user agent). This agent contains modules for getting input from user as well as methods for inter-agent communication. It asks the user for providing their requirements and provides the user with results after mining. The agent stores the history of user interaction, and user profiles with their specific preferences (Giannella C. et al., 2004).

**Facilitator agent:** Facilitator agent is mainly responsible for activation and synchronization. It seeks assignments from interface agents and presents the final results to interface agent.

**Resource agent:** The resource agent actively maintains the Meta data information about each of the data sources. It also provides predefined and ad-hoc retrieval capabilities. It takes into account the heterogeneity of databases.

**Mining agent:** Data Mining agents implement some specific data mining techniques and algorithms. It carries out Data Mining Activity. It captures result of DM and communicates it to result agent or facilitator agent (Giannella C. et al., 2004).

**Result agent:** Result Agent observes a movement of mining agents, and obtains result from mining agents. It stores details about report templates and visualization primitives that can be used to present result to user (Giannella C. et al., 2004).

**Broker agent:** Broker Agent serves as advisor agents that facilitate diffusion of request to agents that have expressed an ability to handle them. It keeps track of names, ontologies and capabilities of all registered agents in the system. It can reply to query of an agent with names and ontology of appropriate agent that has capabilities requested (Giannella C. et al., 2004).

**Query agent:** Query Agent is generated at each demand of a user. The knowledge module contains Meta

data information including local schemas and global schemas. The schemas are used in generating necessary queries for data retrieval.

**Mobile Agent:** Mobile Agents travels around their network. It processes the data and sends result back to main host. The main advantage here is low network traffic. There is also requirement for installing agent platform at each site (Giannella C. et al., 2004).

# 4.1.2 CAKE (Classifying Associating and Knowledge Discovery) architecture

The CAKE architecture is based on centralized Parallel Data Mining Agents (PADMAs). CAKE is a 4-tier architecture where the Distributed Data Mining is implemented using parallel Data Mining Agents (PADMAs) using centralized metadata which contains all the rules of Classification and Association along with its data structure details and web interface used to provide the users with the interface to view the result (Danish Khan, 2008).



Figure-3. CAKE Architecture (M. Klusch et al., 2003).

The PADMAs are going to be executed on the sites where Data Warehouse exists for improving the performance and privacy factor; agents are required only once to retrieve their respective configuration.

PADMAs are combination of Multi-Purpose agents, which are of three major categories according to their respective roles at each stage of Data Mining process:

a) **Rule defined agents:** They used to define the Meta- data of Data Warehouse on the basis of rules that are going to be defined by the users. These rules are combinations of conditions and weighted values, defined to perform the operation for evaluating the data and identify the dependency and relationship between attributes to ascertain the hidden knowledge.

#### www.arpnjournals.com

The rules are going to be used by Intelligent Data Mining Agents and by Knowledge Discovery Agents (Danish Khan, 2008).

- b) **Intelligent data mining agents:** The Data Mining Agents are group of agents, which can setup to work on specified set of data on any location with defined rules. These groups of agents will work together to mine the data and compute the desired result (Danish Khan, 2008).
- c) **Knowledge d iscovery and a gents:** The Knowledge Discovery Agents are used to determine the final computed results in success or failure along with the explanation on computed data. These decisions are taken on the basis of defined requirements in the repository (Danish Khan, 2008).

The Metadata or agent repository is a database that itself used by PADMAs to perform their respective jobs. It contains he data warehouse metadata that is necessary or required by PADMAs. All the agents are to access the Repository once they are being initialized, so they themselves can update their respective tasks assigned in the form of groups created by using "Rule Definer Agents" (Chris Clifton, 2001).



Figure-4. CAKE component diagram.

There are 3 major steps required to be executed in order to complete the data mining process:

Metadata layer: The process of defining the data a) about data which includes data warehouse and its data structure description along with the rules for performing data mining. There is also a detail of remote data warehouse and configure the group description of Data Mining and Knowledge agents. The repository resides on a centralized location from where all information is being fetched by executed agent group according to its rule (http://wwwfp.mcs.anl.gov/ccst/research/reports\_pre19 98/algor ithm\_development/padma/kargupta.html).

b) **Data mining layer:** An agent can be executed from multiple locations to achieve their configured objective. These objectives are defined while configuring the metadata. One type of mining agent performs operation on data while the other one discovers the required result for computed data in warehouse

(http://wwwfp.mcs.anl.gov/ccst/research/reports\_pre1 998/algor ithm\_development/padma/kargupta.html).

c) Presentation layer: The web pages and reports can be developed to provide user friendly results to users. The users can access the mining results from all the sources. Furthermore we can also provide any other OLAP tool which supports the feature of simulating the results as a whole (http://wwwfp.mcs.anl.gov/ccst/research/reports\_pre1 998/algor ithm\_development/padma/kargupta.html).

### 4.2. DDM based on Meta learning: JAM architecture

Meta learning system is implemented by JAM system. JAM is a distributed agent based data mining system. It provides a set of learning agents which are used to compute classifier agents at each site. The launching and exchanging of each classifier agents take place at place at all sites distributed data mining system by providing a set of Meta learning agents which combined the computed models at different sites. JAM is a first system that employs Meta learning as a means to mine distributed databases (Sanjay Kumar Sen et al., 2012).

The Meta learning executes the learning process in parallel on subsets of training data sets which improves efficiency. Executing the same serial program in parallel improves time complexity. In Meta learning is in small subsets of data which can easily accommodate in main memory instead of huge amount of data. Meta learning combines different learning system each having different inductive bias, as a result predictive performance is increased. Meta learning constitutes a scalable machine learning method because it generalizes to hierarchical multilevel Meta learning. Most of these algorithms generate classifiers by applying the same algorithms on different database (Adnan M. Al-Khatib and Ezz Hattab, 2007).

First all local classifiers are computed on each local data site by executing learning agents. Then these local computed classifiers are combined with each local classifiers through Meta learning agents. Each local data sites are administered by local configuration file which is used to perform learning and Meta learning task. After computing base and Meta classifiers JAM system executes the modules for classification of desired data sets. The Configuration File Manger (CFM) is used as server which is responsible for keeping the state of system up to date (Adnan M. Al-Khatib and Ezz Hattab, 2007).

#### www.arpnjournals.com



VOL. 9, NO. 12, DECEMBER 2014

# Figure-5. JAM architecture (Sanjay Kumar Sen *et al.*, 2012).

The above Figure depicts the JAM system with three data sites A, B, C while exchanging their classifier agents. JAM is under distributed protocols in which participating database sites executes independently and also collaborates with other data sites. Data mining system with JAM system solves the problem of how to make a learning system evolve and adjust according to its changing environment. Even though the traditional data mining data mining systems are static which cannot adapt to new system, but data mining system with JAM are adaptive to new environment.

JAM is achieved by employing Meta learning techniques to design learning systems capable of incorporating into their accumulating knowledge the new classifiers that capture patterns which are learner on new data sources (Adnan M. Al-Khatib, 2011).

#### 4.3. DDM based on grid: VEGA architecture

The design process starts by searching and selecting the resources needed to compose the application. The step is accomplished by means of DAS (Data Access Service) and TAAS (Task Access Service) tools that analyze the XML Meta data documents which represent the available resources of the participant K-Grid nodes stored into their KMRs. Meta data about resources are stored in TMRs (Task Management Repositories), a local storage space that contains information about resources selected to perform a computation (Datta S. *et al.*, 2006).



Figure-6. Design process of data mining computation (Mario Cannataro *et al.*, 2004).

The design of data mining computation is performed by Execution Plan Management Service (EPMS). Visual Environment for Grid Applications (VEGA) integrates functionalities of EPMS and other Kgrid services.



Figure-7. Visual Environments for Grid Applications (Mario Cannataro *et al.*, 2004).

VEGA provides following EPMS operations:

- a) Task Composition.
- b) Task Consistency Checking
- c) Execution Plan Generation

Task composition is performed by means of graphical interface which provides a user with a set of



#### www.arpnjournals.com

graphical objects representing grid nodes and resources. Task composition is implemented by software components such as resource manager, object manager, workspace manager. Task consistency checking is to obtain a correct result and consistent model of computation. The validation process is performed by means of two components: the model preprocessor and model postprocessor. In execution plan generation the computation model is translated into an execution plan represented by an XML document. The task is performed by execution plan generator (I. Foster and C. Kesselman, 2001), (http://www.gridforum.org/building\_the\_grid.htm).

## 5. ANALYSIS

In Table-1 we have compared various advantages and disadvantages of each framework.

Type of DDM	DDM Frameworks	Advantages	Disadvantages	
DDM based on	MADM	Easy to build architecture	Agent constrained processing agent-action ability.	
parallel data mining	CAKE	Clear distinction of functionality between agents	Local data sources have restricted availability due to privacy.	
DDM based on Meta learning	JAM	Adaptive learning, Interactive Mining	Learning capability agent needs to be fed up with learning and reasoning algorithm	
DDM based on Grid	VEGA	Improved speed of execution compared to any other data mining algorithm.	Data fusion and preparation are difficult	

Table-1. Analysis of various DDM Architectures its Pros and Cons.

In Table-2 we have analyzed practical application where we can use each of the frameworks.

Type of DDM	DDM Frameworks	Heterogeneous Data Mining	Data Consistency	<b>Communication</b> <b>Cost Reduction</b>	Data Variance	Privacy Preserving
	MADM	Y	Y	N	N	Ν
DDM based on Parallel Data Mining	CAKE	Y	Y	Ν	N	Ν
DDM based on Meta learning	JAM	Y	Y	N	Y	Y
DDM based on Grid	VEGA	N	Y	Y	Y	N

Table-2. DDM frameworks and their preferred applications.

Table-3. Analysis of various DDM frameworks vs major data mining challenges.

Type of DDM	<b>DDM Frameworks</b>	Application		
DDM based on	MADM	Web mining, text mining		
parallel data mining	CAKE	Intrusion Detection, Customer Relationship		
		management, Parallel genetic Algorithm,		
		Financial Data Management		
DDM based on Meta learning	JAM	Business Intelligence, Artificial Immune		
		System, Knowledge management and		
		Marketing, Semantic Web		
DDM based on	VECA	Peer to Peer Computing and Service, E-		
Grid	VEGA	Commerce, Internet and Network Services		

## www.arpnjournals.com

## 6. CONCLUSIONS

In this paper, we classified data mining algorithms into three types: DDM based on parallel data mining, DDM based on Meta learning, DDM based on grid. For each classification we have analyzed the prominent data mining framework, pros and cons of each framework and proposed the most suitable application where each framework can be most appropriate. Finally, we analyzed the paper using important parameters required for a good distributed data mining algorithm.

## REFERENCES

- Adnan M. Al-Khatib. 2011. Detect CNP Fraudulent Transactions. World of Computer Science and Information Technology Journal (WCSIT) ISSN: 2221-0741. 1(8): 326-332.
- [2] Adnan M. Al-Khatib. 2007. Mining Fraudulent Behavior in e-payment Systems. Ph.D. Dissertation.
- [3] Adnan M. Al-Khatib and Ezz Hattab. 2007. Mining Fraudulent Transactions in e-payment Systems. 9<sup>th</sup> international conference IIWAS 2007. pp. 179-189.
- [4] Bin Liu, Shu-Gui Cao, Xio-Li Jia, Zhao-Hua-Zhi. 2010. Data Mining in Distributed Environment. Proceedings of Ninth International Conference on Machine Learning's and Cybernetics Qingdao. pp. 421-427.
- [5] 2000. Building the Grid: An Integrated Services and Toolkit Architecture for Next Generation Networked Applications, Foster. [Online]. Available: http://www.gridforum.org/building\_the\_grid.htm.
- [6] Chris Clifton. 2001. Privacy Preserving Distributed Data Mining. Department of Computer Sciences.
- [7] Dam. H. H., Abbass. H. A. and Lokan. C. 2005. DXCS: an XCS system for distributed data mining. Proceeding of the 2005 Conference on Genetic and evolutionary computation, Washington, DC, USA. pp. 1883-1890.
- [8] Danish Khan. 2008. CAKE- Classifying, Associating and Knowledge Discovery: An Approach for Distributed Data Mining (DDM) Using Parallel Data mining Agents (PADMAs). IEEE ACM International Conference on Web Intelligence and Intelligent Agent Technology. pp. 596-602.
- [9] Datta S., Bhaduri K., Giannella C., Wolff R. and Kargupta H. 2006. Distributed learning with bagginglike performance, Pattern Recognition Letters 24, 455-471. 2006. Distributed data mining in peer-to-peer networks, IEEE Internet Computing. 10(4): 18-26.

- [10] Giannella C., Bhargava R. and Kargupta H.m. 2004. Multi-agent systems and distributed data mining. In Cooperative Information Agents VIII: 8th International Workshop, CIA 2004, Erfurt, Germany. pp. 1-15.
- [11] I. Foster and C. Kesselman. 2001. The anatomy of the grid: Enabling scalable virtual organizations. S. Tuecke. International Journal Supercomputer Applications. 15(3).
- [12] Mario Cannataro, Antonio Conguista, Andrea Puglesea, Dominico Talia, Paolo Trunfio. 2004. Distributed Data Mining on Grids: Services, Tools and Applications. IEEE Transactions on Systems, MAN and Cybernetics-part B Cybernetics. 4(6).
- [13] M. Klusch, S. Lodi, G. Moro. 2003. Agent-based Distributed Data Mining: The KDEC Scheme. Intelligent Information Agents - The Agent Link Perspective. Lecture Notes in Computer Science 2586 Springer.
- [14] Parallel Data Mining Agents (PADMA) http://wwwfp.mcs.anl.gov/ccst/research/reports\_pre19 98/algorithm\_development/padma/kargupta.html.
- [15] Prodromidis A. L., Chan P. K. and Stolfo S. J. 2000. Meta-learning in distributed data mining systems: Issues and approaches, In Advances in Distributed and Parallel Knowledge Discovery, pp. 81-114. The MIT Press.
- [16] Sanjay Kumar Sen, Dr. Sujata Dash, Subrat P Pattanayak. 2012. Agent based Meta learning in distributed data mining system. International Journal of Engineering Research and Applications (IJERA). 2(3): 342-348.