Sanjay Kumar Sen Int. Journal of Engineering Research and Applications www.ijera.com
ISSN : 2248-9622, Vol. 4, Issue 6(Version 6), June 2014, pp.35-38

RESEARCH ARTICLE OPEN ACCESS

Integration of JAM and JADE Architecture in Distributed Data Mining System

Sanjay Kumar Sen, Dr. Subhendu Kumar Pani.
Associate Professor (Comp Sc & Engg.), OEC, Bhubaneswar.
Associate Professor (Comp Sc & Engg.), OEC, Bhubaneswar.

ABSTRACT
Data mining systems is used to discover patterns and extract useful information from facts recorded in databases. Knowledge can be acquired from database by using machine learning algorithm which compute descriptive representations of the data as well as patterns that may be exhibited in the data. Most of the current generation of learning algorithms, however, are computationally complex and require all data to be resident in main memory which is clearly untenable for many realistic problems and databases. The main focus of this work is on the management of machine learning programs with the capacity to travel between computer sites to mine the local data. This paper describes the system architecture of JAM (Java Agents for Meta-learning), a distributed data mining system that scales up to large and physically separated data sets. In a single repository data base where data is stored in central site, then applying data mining algorithms on these data base, patterns are extracted, which is clearly implausible and untenable for many realistic problems and databases. To deal with these complex systems has revealed opportunities to improve distributed data mining systems in a number of ways.This paper describes the system architecture of JAM (Java Agents for Meta-learning), a distributed data mining system that scales up to large and physically separated data sets JAM is an extensible agent-based distributed data mining system that supports dispatch and exchange of agents among participating data sites and meta-learning techniques to combine the multiple models that are learned. A brief description of JADE architecture is also given.
***Keywords**:* compatibility, distributed data mining, global classifier

I. Introduction

Data mining systems aim to discover patterns and extract useful information from facts recorded in databases[2]. Applying various machine learning algorithms which compute descriptive representations as well as patterns from which various knowledge can be acquired. However, are computationally complex and require all data to be resident in main memory which is clearly untenable for many realistic problems and databases[2]. Traditional data analysis methods that require humans to process large data sets are completely inadequate. Applying the traditional data mining tools to discover knowledge from the distributed data sources might not be possible [4]. Therefore knowledge discovery from multi-databases has became an important research field and is considered to be a more complex and difficult task than knowledge discovery from mono-databases [8]. The relatively new field of Knowledge Discovery and Data Mining (KDD) has emerged to compensate for these deficiencies. Knowledge discovery in databases denotes the complex process of identifying valid, novel, potentially useful and ultimately understandable patterns in data [1]. Data mining refers to a particular step in the KDD process. According to the most recent and broad definition [1], “data mining consists of particular algorithms (methods) that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns (models) over the data.” A common methodology for distributed machine learning and data mining is of two-stage, first performing local data analysis and then combining the local results forming the global one [5]. For example, in [6], a meta-learning process was proposed as an additional learning process

II. Meta-learning

Meta-learning[10] is loosely defined as learning from learned knowledge. Meta-learning is a recent technique that seeks to compute higher level models, called meta-classifiers, that integrate in some principled fashion the information cleaned by the separately learned classifiers to improve predictive performance. In meta learning process a number of learning programme is executed on a number of data subsets in parallel then collective result is collected in the form of classifiers.

2.1. Meta-learning Techniques

1.Using a variety of statistical, information-theoretic and the characterization of datasets is performed.

2. By applying a set of algorithms at the base level and combining these through a meta learner information is extracted.
3. To accelerate the rate of learning process, knowledge is extracted through a continuous learner.

In meta-learning (learning from learned knowledge) technique dealing with the problem of computing a global classifier from large and inherently distributed databases. A number of independent classifiers – "base classifiers" -are computed in parallel. The base classifiers are then collected and combined to a „meta-classifier" by another learning process. Meta-classifiers can be defined recursively as collections of classifiers structured in multi-level trees [9]. Such structures, however, can be unnecessarily complex, meaning that many classifiers may be redundant, wasting resources and reducing system throughput.

1. The JAM Architecture:

By executing learning agents all local classifiers are computed. All local computed classifiers are exchanged between local sites and combine with Then these local computed classifiers are exchanged between local sites combine with each local classifiers through meta-learning agents. Each local data sites is administered by local configuration file which is used to perform the learning and meta-learning task. To supervise agent exchange work and execution of meta-learning process smoothly, each data site is employed by GUI and animation facilities of JAM[2]. After computing the base and meta-classifiers, the JAM system executes the modules for classification of desired data sets. The configuration file manager(CFM) is used as server which is responsible for keeping the state of the system up to date[3].

III. Advantages of Meta-learning

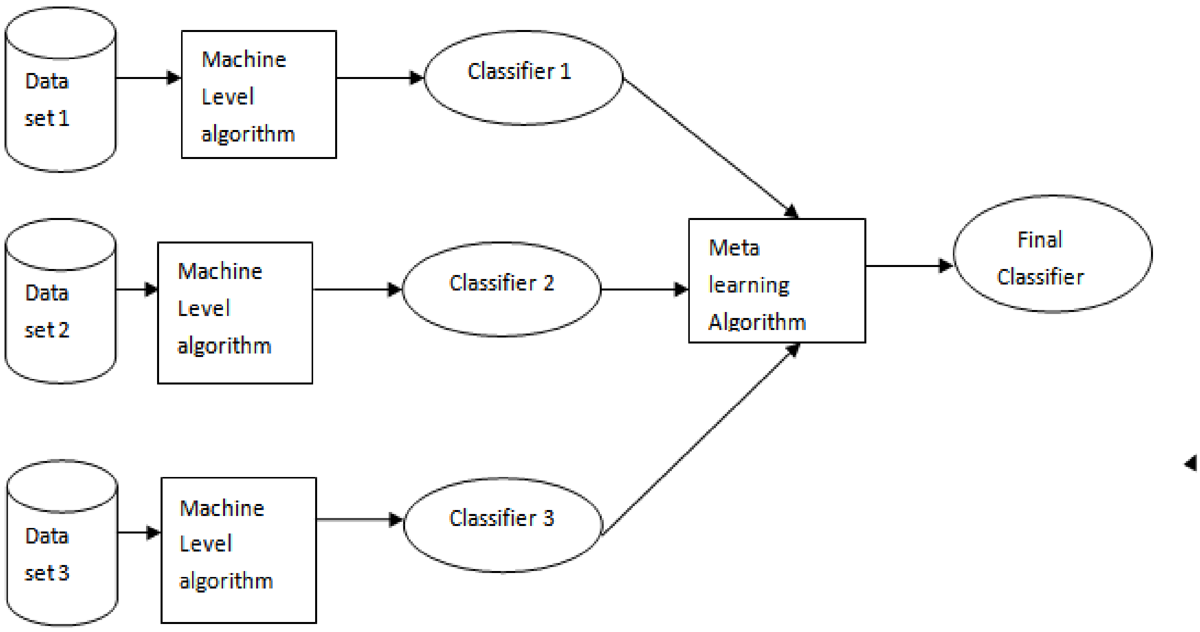
The same base learning process is executed in parallel by meta-learning on subsets of the training data set which improves efficiency[2]. Because the same serial programme is executed in parallel which improves time complexity. Another advantage is that learning is in small subsets of data which can easily accommodated 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. A higher level learned model is derived after combining separately learned concepts. Meta-learning constitutes a scalable machine learning method because it generalizes to hierarchical multi-level meta-learning. Also most of these algorithms generate classifiers by applying the same algorithms on different data base.

i. Scalability

The data mining system is highly scalable because its performance does not hamper as the data sites increases[2]. It depends on the protocols that transfer
and manage the intelligent agents to support the data sites.

ii. Efficiency

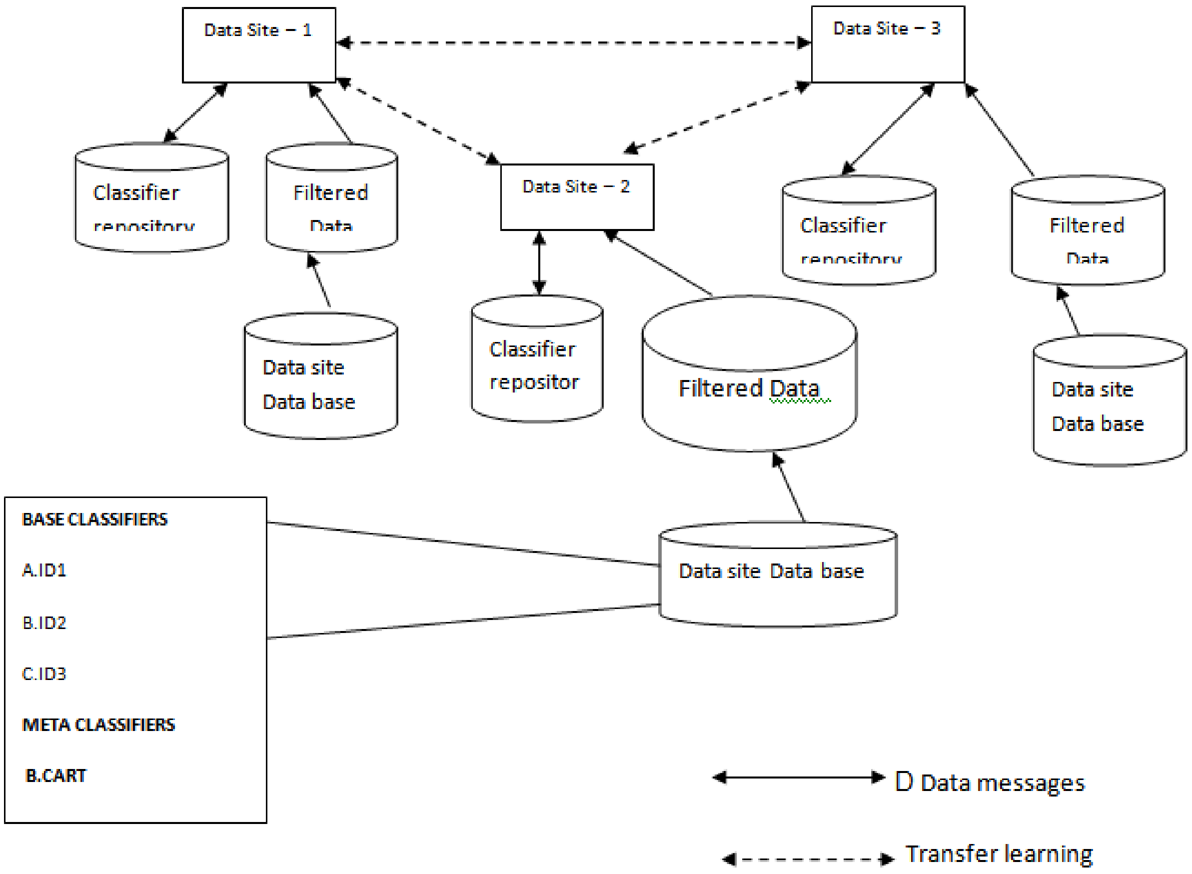
It refers to the effective use of the available system resources. It depends on
the appropriate evaluation and filtering of the available agents which minimizes redundancy[2]



From Figure 1, the different stages of a simplified meta-learning scenario are

JAM (Java Agent for Meta-Learning) :

Meta-learning system is implemented by JAM system(Java Agents for Meta-learning) which is a distributed agent-based data mining system[2]. It provides a set of learning agents which are used to compute classifier agents at each site[2]. The launching and exchanging of each classifier agents takes place at all sites of distributed data mining system by providing a set of meta-learning agents which combined computed models those computed models at different sites[2]. This goal can be achieved through the implementation and demonstration of a system we call JAM (Java Agents for Meta-Learning). To our knowledge, JAM is the first system to date that employs meta-learning as a means to mine distributed databases. A commercial system based upon JAM has recently appeared [3].



& classifier agents

The JAM ARCHITECTURE

2. JADE(Java Agent Development Environment)

JADE (Java Agent DEvelopment Framework) is a software framework to make easier the development of agent applications in compliance with the FIPA specifications for interoperable intelligent multi-agent systems. The goal of JADE is to simplify development while ensuring standard compliance through a comprehensive set of system services and agents.

JADE architecture includes the followings components

Agent It executes tasks and interaction takes place through exchange of. Messages.

Container It can executed on different hosts thus achieves a distributed platform. Each container can contain zero or more agents.

- Platform – It provides message delivery.
- Main Container- The main container is itself a container and can therefore contain agents, but differs from other container as it must be the first container to start in the platform and all other containers register to it at bootstrap time.
- AMS and DF- It represents the authority in the platform and is the only e agent able to perform platform management actions such as starting and killing agents or shutting down the whole platform. The DF that provides the Yellow pages service where agents can publish the service they provide and find other agents providing the services they need.

JADE is written in Java language and is made by various Java packages, giving application programmers both ready-made pieces of functionality and abstract interfaces for custom, application dependent tasks[9]. Java was the programming language of choice because of its many attractive features, particularly geared towards object-oriented programming in distributed heterogeneous environments; some of these features are Object Serialization, Reflection API and Remote Method Invocation (RMI)[9].

Reference

[1] R. Grossman, S. Baily, S. Kasif, D. Mon, and A. Ramu. *The preliminary design of papyrus: A system for high performance*. In P. Chan H. Kargupta, editor, Work. Notes KDD-98 Workshop on Distributed Data Mining, pages 37–43. AAAI Press, 1998.

[2] Sanjay Kumar Sen, Dr Sujata Dash, Subrat P Pattanayak "*AGENT BASED META LEARNING IN DISTRIBUTED DATA MINING SYSTEM*". International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp. 342-348.

[3] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in Knowledge Discovery and Data Mining*. AAAI Press/MIT Press, Menlo Park, California/Cambridge, Massachusetts / London, England, 1996.

[4] Kargupta H., Park B., Hershberger D., Johnson E., *Collective Data Mining: A New Perspectiv Toward Distributed Data Analysis*. Accepted in the Advances in Distributed Data Mining, H. Kargupta and P. Chan (eds.), AAAI/MIT Press, (1999).

[5] Zhang X., Lam C., Cheung W.K., *Mining Local Data Sources For Learning Global Cluster Model Via Local Model Exchange*. IEEE ntelligence Informatics Bulletin, (4) 2 (2004).

[6] Prodromidis A., Chan P.K., Stolfo S.J., *Meta-learning in Distributed Data Mining Systems:Issues and Approaches*. In: Advances in Distributed and Parallel Knowledge Discovery, Kargupta H., Chan P.(ed.), AAAI/MIT Press, Chapter 3, (2000).

[7] Liu H., Lu H., Yao J., *Identifying Relevant Databases for Multi database Mining*. In: Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining, (1998) 210.

[8] Ahang S., Wu X., Zhang C., *Multi-Database Mining*. IEEE Computational Intelligence Bulletin, (2)1 (2003).

[9] Fabio Bellifemine, Agostino Poggi, Giovanni Rimassa, "*JADE – A FIPA-compliant agent framework*"

[10]. C. Phua, V. Lee, K. Smith, and R. Gayler, "*A Comprehensive Survey of Data Mining-Based Fraud Detection Research*," http://www.bsys.monash.edu.au/people/cphua/, Mar. 2007.