International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 5, May-2014, pp: (426-431), Impact Factor: 1.252, Available online at: www.erpublications.com

Multi Agent Approach for Evolving Data Mining in Parallel and Distributed Systems using Genetic Algorithms and Semantic Ontology

K. Syed Kousar Niasi[1], Dr. E. Kannan[2]
[1]Assistant Professor in Computer Science, Jamal Mohamed College, Trichy
[2]Dean, Vel Tech DR. RR & DR. SR Tech University, Chennai

Abstract: Mining information from parallel and distributed systems has been discussed in wide spectrum, and the earlier approaches suffer with the result produced for any query with missing values. Also the approaches have more time complexity which reduces the throughput of the overall systems. We propose a new multi agent based approach for mining information from large distributed systems, where the data scattered throughout the network in different systems. Unlike earlier approach the proposed method generates number of agents according to the availability of data and generates results based on the fitness function designed at genetic algorithm. The proposed approach maintains an agent container at each of the location and each has Meta data about the data locations and information. The meta data are stored in form of semantic ontology in order to reduce the data look up time and reduce the time complexity. Upon query submission, the proposed method identifies set of locations where the information is available and generates number of agents to fetch the information from different nodes of the network. The retrieved results are evaluated with genetic algorithm for relevancy of result toward the query submitted. The proposed approach produces efficient results in accuracy of results and time complexity.

Keywords: Semantic Ontology, Multi Agent Systems, Data Mining, GA, Parallel and Distributed Systems.

Introduction

The growth of internet technology and information has increased the necessary of maintaining information in distributed systems where the information could be accessed from different location based on the location of user. The number of locations and the volume of data decide the efficiency of the retrieval system in time complexity and efficiency of result produced. Whenever the information stored in distributed systems, the look up procedure must be efficient, so that the location and availability of data can be identified easily. The earlier approaches has used many data representation and storage schemes like range trees. The problem with earlier approaches is the look up time of data is more and that increases the time complexity of overall system. Also in parallel and Distributed Systems, the query submitted has to be executed in different machines at the same time in order to reduce the execution time. The multi agents system is the platform where we can maintain any number of agents. Each agent has the behavior of mobility, that they can move from one location to another where the data is available. Using these agents we can perform query execution at multiple locations at the same time to reduce the time complexity of the system designed. The Java Agent Development Environment (JADE) is the platform which provides mobility of agents and can be controlled using Agent Control Messages. Using the language the agent can be moved from locations to other easily and they work based on events generated.

Data mining is the process of mining information from large volume of data set. The data may be present in different location, but the data mining algorithm has to retrieve the exact relevant information from different locations to produce efficient results. There are many approaches available for data mining; we use mobile agents to retrieve the information from different locations of the network. The genetic algorithm which uses cross over and mutation operations to select the information extracted and has a fitness function to evaluate the selection process. We apply the genetic algorithm to evaluate the retrieval process. Semantic ontology is a relational representation of information, where the ontology consists of classes and labels. For each word there are synonyms or relational terms like hotel: Boarding: Lodging, the ontology file consists of classes, labels and similar meanings about a concept. In our case the ontology file consists of classes and related terms about the data and locations. Using these functionalities we propose a new mining approach to increase the efficiency of the retrieval systems.

Related Works

There exists various approaches for mining information from parallel and distributed systems; we discuss few of them around the problem identified. Data Partitioning and Association Rule Mining Using a Multi-Agent System [9], explores and demonstrates (by experiment) the capabilities of Multi-Agent Data Mining (MADM) System in the context of parallel and distributed Data Mining (DM). The exploration is conducted by considering a specific parallel/distributed DM scenario, namely data (vertical/horizontal) partitioning to achieve parallel/distributed ARM.

To facilitate the partitioning a compressed set enumeration tree data structure (the T-tree) is used together with an associated ARM algorithm (Apriori-T). The aim of the scenario is to demonstrate that the MADM vision is capable of exploiting the benefits of parallel computing; particularly parallel query processing and parallel data accessing. In addition the approach described offers significant advantages with respect to computational efficiency when compared to alternative mechanisms for (a) dividing the input data between processors (agents) and (b) achieving distributed/parallel ARM.

Multiagent System for Pattern Searching in Billing Data [10], present an agent-based pattern searching system using a distributed Apriori algorithm to analyse billing data. In the paper, we briefly present the problem of pattern mining. Next, we discuss related research focusing on distributed versions of Apriori algorithm and agent-based data mining software. Paper continues with an explanation of architecture and algorithms used in the system. We propose an original distribution mechanism allowing to split data into smaller chunks and also orthogonally distribute candidate patterns support calculation (in the same computation task). Experimental results on both generated and real-world data show that for different conditions other distribution policies give better speedup. The system is implemented using Erlang and can be used in heterogeneous hardware environment. This, together with multi-agent architecture gives flexibility in the system configuration and extension.

A Multi-Agent System for Context-Based Distributed Data Mining [11], describes an approach that aims to resolve this issue. Focusing on a key business problem -- the prediction of customer behaviour -- it presents a distributed multi-agent framework that deals with context heterogeneity via hierarchical modeling. The main elements of this work are to (1) provide a solution to the contextual heterogeneity problem in distributed data mining and (2) design and implement a hybrid distributed system for the proposed distributed data mining approach. Extracting Peculiar Data from Multidatabases Using Agent Mining [12], discusses the peculiar data mining and agent mining. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Agent Mining:

The Synergy of Agents and Data Mining [13], give an overall perspective of the driving forces, theoretical underpinnings, main research issues, and application domains of this field, while addressing the state-of-the-art of agent mining research and development. Our review is divided into three key research topics: agent-driven data mining, data mining-driven agents, and joint issues in the synergy of agents and data mining. This new and promising field exhibits a great potential for groundbreaking work from foundational, technological and practical perspectives.

A Bounded and Adaptive Memory-Based Approach to Mine Frequent Patterns From Very Large Databases [8], could use only a bounded portion of the primary memory and this gives the opportunity to assign other parts of the main memory to other tasks with different priority. In other words, we propose a specialized memory management system which caters to the needs of the ARM model in such a way that the proposed data structure is constructed in the available allocated primary memory first. If at any point the structure grows out of the allocated memory quota, it is forced to be partially saved on secondary memory. The secondary memory version of the structure is accessed in a block-by-block basis so that both the spatial and temporal localities of the I/O access are optimized. Thus, the proposed framework takes control of the virtual memory access and hence manages the required virtual memory in an optimal way to the best benefit of the mining process to be served. Several clever data structures are used to facilitate these optimizations.

All the above discussed methods have the problem of data retrieval and its accuracy about time complexity.

Proposed Method

The proposed multi agent system has three components namely Lookup manager, Agent Scheduler and GA Based Evolution. The lookup manager identifies set of locations where the data is available, the agent scheduler generates number of agents and moves them to remote locations to perform the query operation, and finally the GA evolution process verifies the retrieved data about the query.

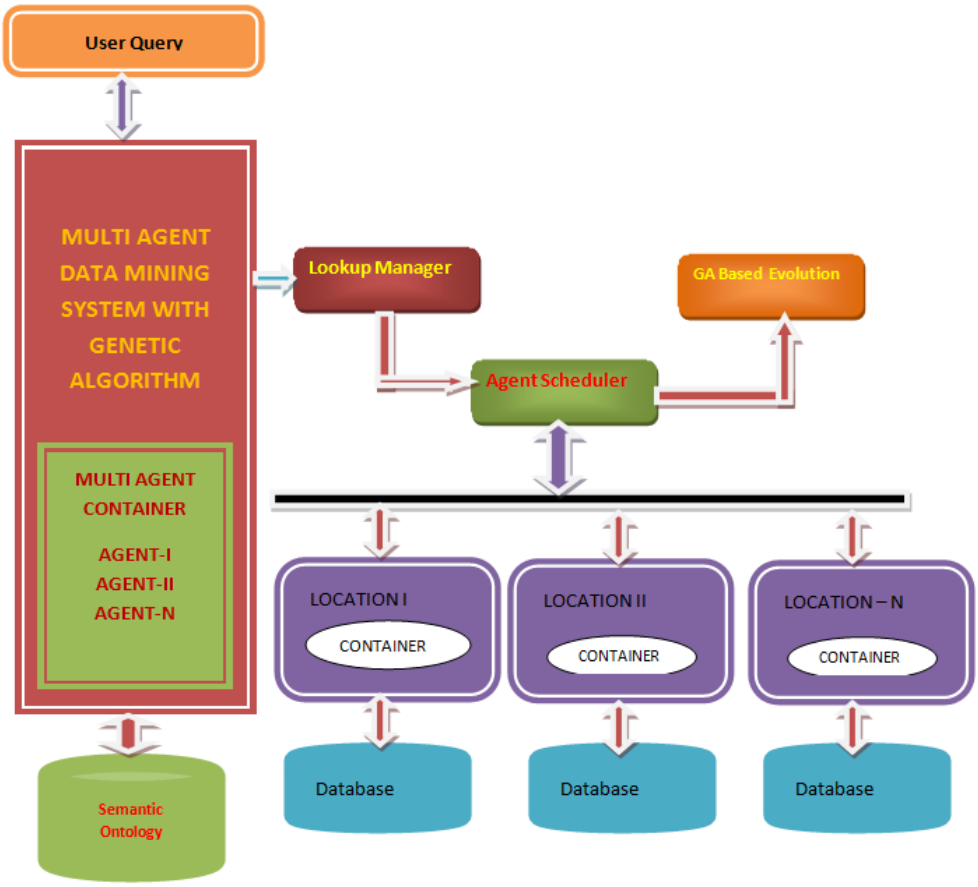


Figure 1: Proposed System Architecture

A. Lookup Manager

The input query is processed to identify the set of data objects necessary to process the query. From identified objects, the location of objects available are identified using the semantic ontology. The semantic ontology consists the meta information about the data objects present in distributed networks and based on the meta information, a set of locations are selected where the original data is available.

Algorithm:

Input: Search Query sq.
Output: set of data locations Dl.

step1: Read ontology set O.
Step2: Identify keywords and their location and substring the query.
Sqs = substring(sq, keywords).
Step3: $Ts = \Sigma \text{ Terms} \in sqs$
Step4: Extract data object name from sq.
$DO = \int_{i=0}^{N} \sum T \in Ts$
Ts = set of terms appears after 'from' clause.
Step4: for each Term T_i of Ts
$Dl = \Sigma Dl + \int_{i=0}^{size\ (O)} Ti \in Class(i)$
end.
step5: stop.

B. Agent Scheduler

The identified locations are selected as input to this process, based on the location we compute the number of locations where the data is available. Based on number of locations, N number of agents will be generated and initialized with the query parameters and location information. Once the agents are generated and initialized , all of them will be scheduled to move to remote location in same time. When the agent moves to the remote location, it performs the query execution there and comes back with the result. When all the agents returns to the home container, it updates the shared memory where the query results are stored.

Algorithm:
Input: search query sq, location set Dl.
Output: Results set Rs.
Step1: compute number of locations Nl = Σloc(Dl).
Step2: for each location l of Nl
Agent ma = Create new Agent.
ma.location = l.
ma.query = sq.
Add agent to home container Hcon = ΣAgents + ma.
end.
Step3: for each agent A_i of Hcon
generate event and move to remote location.
end.
Step4: if location==Remote
perform query operation.
move to home container.
end.
Step5: if location == home
update shared memory Rs = ΣRecords + Result set.
end.
step6: stop.

C. GA Evolution

The genetic algorithm approach is used to verify the resultant data and the input query. The result produced should be relevant and we must generate concrete result to the user. There are many functions exists to compute the fitness function, we use support function to calculate the fitness values. Based on the fitness value the irrelevant results are identified. The fitness value is computed only once for each node. From generated results we compute the average relevancy with each tuple selected. The average relevancy must be greater than relevancy threshold which is maintained by the fitness function. Then cross over and mutation operations will be performed.

Algorithm:
Input: Result set Rs.
Output: Final results Fr.
step1: Initialize data points with Rs.
Step2: for each data point Rs_i from Rs
perform cross over.
perform mutation.
compute fitness value fv = $\int_{i=0}^{M} \text{Count}(\forall(\text{Conditions} \in \text{query}) \in \text{Rs})$
if fv> fitness threshold
add Rs_i to final results.
Fr = ΣRs_i(Fr)+ Rs_i
else
drop Rs_i.
end
end.
Step3: stop.

Results and Discussion

The proposed multiagent based data mining technique using genetic algorithm for parallel and distributed systems has been implemented using Java agent development environment with 300 locations and containers. We have tested the proposed approach with various numbers of locations and various size of data locations. The proposed approach has produced efficient result both in time complexity and result efficiency.

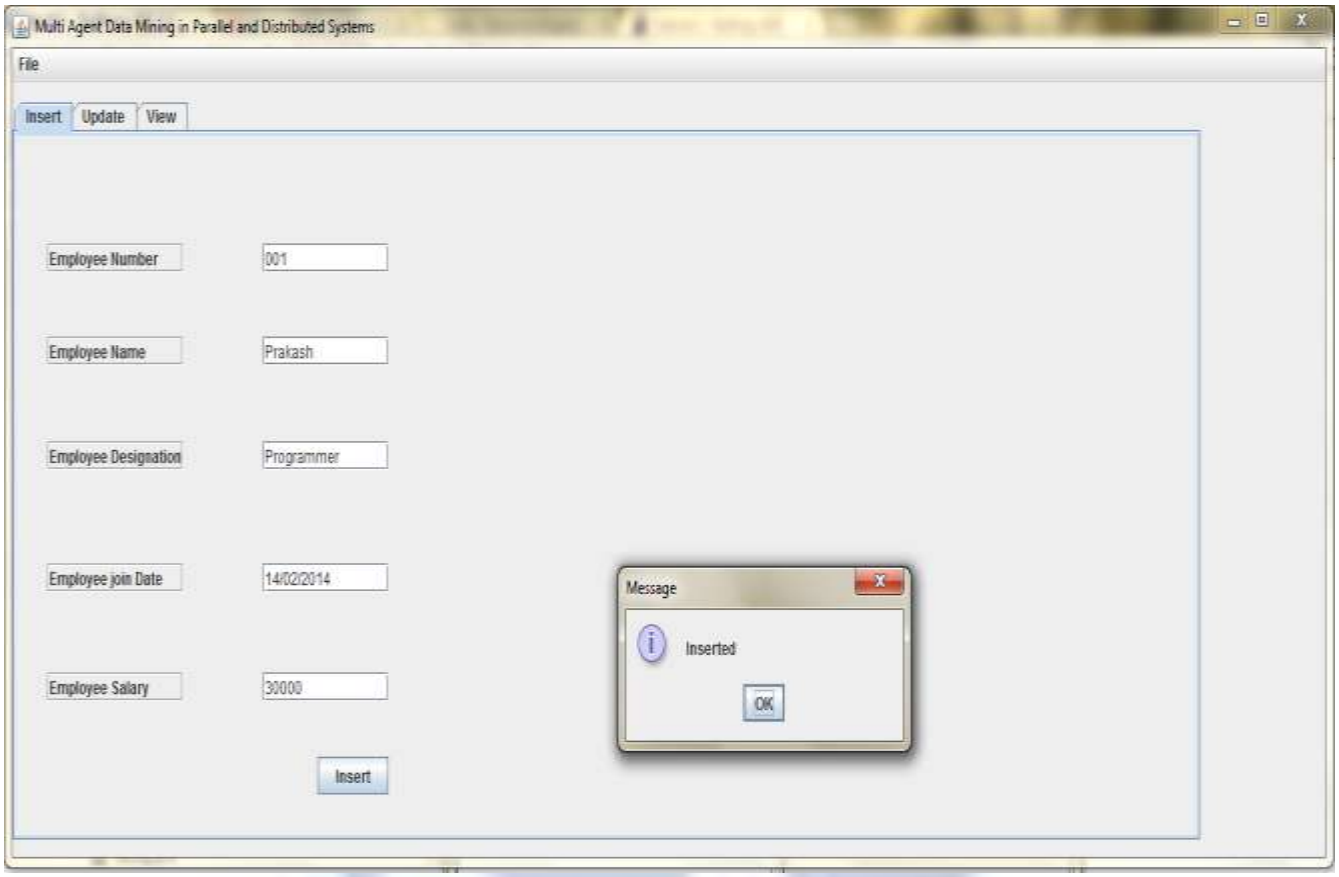


Figure 2: snapshot of result produced by proposed method.

The figure2, shows the snapshot of result produced by the proposed method, where the data is updated in different locations of distributed network using number of mobile agents.

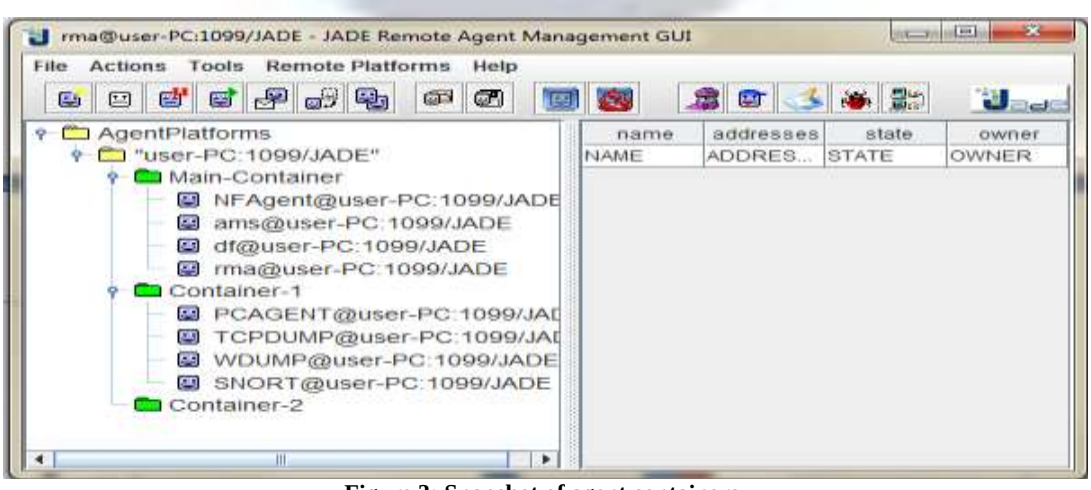
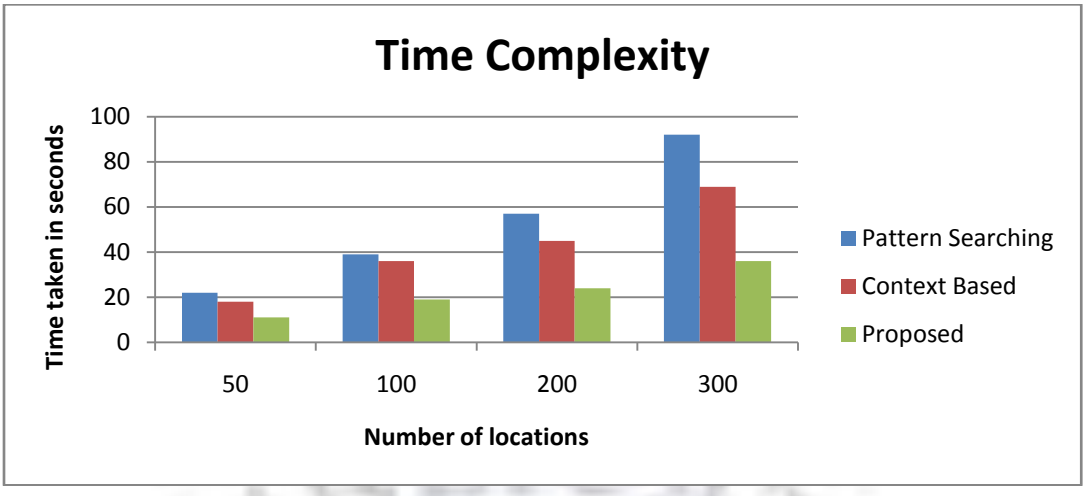


Figure 3: Snapshot of agent containers

The figure 3, shows the snapshot of agent container and number of agents available at the container of the proposed system.



Graph1: shows the time complexity of different approaches

The graph1 shows the time complexity produced by different algorithms, it shows that the proposed approach has less time complexity than other pattern searching and context based approaches.

Conclusion

We proposed a naval multi agent approach for data mining in parallel and distributed systems using genetic algorithm. The method has process the query text to identify the data objects necessary to complete the query. For identified data objects, set of locations are identified using semantic ontology and number of agents will be created. The agents are initialized and moved to remote location to perform the query operation and moved back to home container. The retrieved results are applied with genetic algorithm to get exact and accurate and relevant results. The proposed method has produced efficient and accurate results with low time complexity.

References

[1]. S. Nirmal Chander, P. Ram Prasath, V. Santhosh Kumar, "Enhancing the Relevance of Semantic Web information Retrieval Results using Extention Theory", TISC, 2010.
[2]. Vuda Sreenivasa Rao, Dr. S Vidyavathi, "DISTRIBUTED DATA MINING AND MINING MULTI-AGENT DATA", (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 04, 2010, 1237-1244.
[3]. S.VEERAMALAI, A.KANNAN, "An Intelligent Association Rule Mining Model for Multidimensional Data Representation and Modeling", International Journal of Engineering Science and Technology Vol. 2(9), 2010, 4388-4395.
[4]. Vuda Sreenivasa Rao, S Vidyavathi, G.Ramaswamy, " DISTRIBUTED DATA MINING AND AGENT MINING INTERACTION AND INTEGRATION: A NOVEL APPROACH", IJRRAS 4 (4) , September 2010.
[5]. Vuda Sreenivasarao, Rallabandi Srinivasu, Prof. G.Ramaswamy, Nagamalleswara Rao Dasari, Dr. S Vidyavathi, "The Research of Distributed Data Mining Knowledge Discovery Based on Extension Sets", International Journal of Computer Applications (0975 – 8887), Volume 8– No.2, October 2010.
[6]. Yoones Asgharzadeh Sekhavat, Mohammad Fathian, Mohammad Reza Gholamian, Somayeh Alizadeh, " Mining important association rules based on the RFMD technique", Int. J. Data Analysis Techniques and Strategies, Vol. 2, No. 1, 2010.
[7]. Longbing Cao, Senior Member, IEEE, Yanchang Zhao, Member, IEEE, Huaifeng Zhang, Member, IEEE, Dan Luo, Chengqi Zhang, Senior Member, IEEE, and E.K. Park," Flexible Frameworks for Actionable Knowledge Discovery", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 22, NO. 9, SEPTEMBER 2010.
[8]. Muhaimenul Adnan and Reda Alhajj, "A Bounded and Adaptive Memory-Based Approach to Mine Frequent Patterns From Very Large Databases", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, VOL. 41, NO. 1, FEBRUARY 2011.
[9]. kamal ali anshari, Data Partitioning and Association Rule Mining Using a Multi-Agent System, International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 5, September 2013.
[10]. Lucauze Bevan, Multiagent System for Pattern Searching in Billing Data, Multimedia Communications, Services and Security Communications in Computer and Information Science Volume 368, 2013, pp 13-24.