An Intelligent Agent based Data Preprocessing Software

Sharon Christa
Dept of ISE,
DSCE, Bangalore.

Suma. V
RIIC
DSI, Bangalore

Lakshmi Madhuri
RIIC
DSI, Bangalore

ABSTRACT

With the evolution of distributed computing, the databases were inherently distributed across the globe. The core need in the current industrial environment is to extract information from the huge, complex and dynamic data through data mining techniques. Existence of an inconsistency in the data will directly affect the data mining and thereby affect the business performance. Thus, agents which are a powerful technology for the analysis design and implementation of autonomous intelligent systems is used to handle the varied issues related to inconsistencies in the data. This paper provides the design and development of intelligent software that uses agents to handle the data preprocessing thereby improving and enhancing the quality of data to be mined.

General Terms

Data Preprocessing, Intelligent Agents, Data Mining.

Keywords

Coordinator Agent, Transformation Agent, Discretization Agent.

1. INTRODUCTION

The availability of data storing and management applications evolved in the 1990's and the whole idea of it was taken from the account management system dated back to Mesopotamia civilization. The need for data management originated in 1980's and the progression in technology enables organizations to store, process and update huge and complex data dynamically. Thus, massive amount of data should be handled by the organization in every minute. Unless there is any benefit from the data stored it is useless. Analysing and managing stored data and getting information from it is one of the core requirements of the organization for their growth and improved business performance. In real world data stored can never be perfect due to various reasons like transmission errors, incorrect data, and error while entering data to database, inconsistent formats for input fields, equipment malfunction, and modifications in data was ignored, and corresponding field was not considered important at the time of filling. Existence of an inconsistency in the data will directly affect the data mining and thereby affect the business performance. Thus, eighty percent of the data management time is spent on overcoming the inconsistencies in the data. Data preprocessing is the initial process in data management performed to remove the inconsistencies present in the data.

Even though data management migrated from manual to semi automated and subsequently advanced to fully automated, there are still processes in the data management that requires automation which is the handling of inconsistent data in the database and data warehouses.

This paper provides an overview, design and implementation of intelligent data preprocessing software which can perform all the operations in an automated way. These intelligent agents will themselves perform all data preprocessing activities in lieu of manual setting of parameters and analyzing the data which is the current day scenario in all IT industry.

The organization of the paper is as follows: Section 2 is about the related work done by several research scholars in this domain. Section 3 provides explanation the Rationale of the Architecture. Section 4 depicts the implementation and Section 5 provides brief description about the result analysis and the limitations in the newly developed system. Section 6 gives the summary of the entire work.

2. RELATED WORK

Data in the real world is incomplete, inconsistent and noisy hence, data stored in database often results in mistakes such as out of range values, missing values, impossible data combinations etc [1]. Furthermore, some issues include integration of data from various data sources, use of incomplete data for mining which results in inefficient time consumption and unreliable results [2]. Author in [3] therefore recommends implementation of efficient data preprocessing system to overcome the aforementioned issues. Data preprocessing is a part of data management that has multiple sub processes to deal with various issues in the data [4]. Data preprocessing includes data cleaning, data integration, data transformation, and data reduction. Data cleaning activity eliminates the noise present in the data set. Data integration phase enables merging of data from multiple sources into a coherent data source. Data transformation activity transforms the data into a suitable form for data mining and data reduction activity reduces the data size [5].

Petteri Nurmi et.al [6] present an architecture for distributed data preprocessing in ubiquitous environments, which supports the full distribution of processing tasks and enables the encapsulation of privacy and security mechanisms within every component. Sung Wook Baik et. al [7] presents an agent-based distributed data mining approach dealing with heterogeneous databases located at different sites. It introduces a modified decision tree algorithm on an agent based framework, which produces an accurate global model without transferring data between agents. The novel approach is evaluated with a test case of 184 aerial photograph images. Chunsheng Li and Yatian Gao [8] a data mining model which is based on multi-agent technique to mine the discredit activities in the water sale public service. The Multilayer feed-forward neural network (MFNN) trained by the improved back-propagation (BP) algorithm and decision tree algorithm have been employed in the model. Five agents have been developed to mine the discredit patterns and evaluate the users' credit according to the model.

Based on the research work done by the above authors and the result obtained, agents can be integrated to processes to make it overcome its drawbacks. As mentioned in the introduction, data preprocessing is the process in data management that still needs to be automated. Agents comprise a powerful technology for the analysis, design and implementation of autonomous intelligent systems that can handle distributed problem-solving, cooperation, coordination, communication, and organization in a multiplayer environment [9]. Thus, agents can be used in automating the data preprocessing phase. Agents can handle missing data, noisy data, outliers etc. Also, it can transform data to suitable format for mining. Agents further aid in attribute reduction and dimension reduction [10]. According to Maes, an Agent is software entity that can be used to perform the operations independently and therefore can be used in lieu of user or another program [11]. Each agent has specific characteristics which vary depending on the problem domain. In a multi agent system agents communicate, co-operate and co-ordinate with the other agents. Each agent in the system acts autonomously, and co-operates with other agents. They work together for the tasks to be performed to achieve the goal of the system [12]. By integrating agent technology with data mining the performance of data mining tool further improves. Therefore, integration of agent in data preprocessing is considered to be a sensible approach [13].

3. RATIONALE OF THE ARCHITECTURE

The essential characteristic for an agent based pre-processing system is autonomy, adaptive, reactivity and learning cooperatively. They also need to have the ability to perform functions such as maintaining a pre-processing dictionary to define the structure of raw data given by the user, maintaining a profile of user with respect to data pre-processing, to have ability to clean up the newly updated data and to maintain it in a temporary file, additionally to give suggestions regarding the preprocessing techniques that can be performed on unprocessed data [14]. However, the data to be processed needs to be maintained in a text file. The basic cleaning process is performed after examining the data. Consequently, the structure of data is identified with types of the attributes such as integer or string, nominal or ordinal such that nominal data requires further categorization. Thus, the user can update the data and also can validate the structure of data [15].Figure 1 depicts the phases of data pre-processing using agents. The architecture comprises of five agents namely coordinator agent, discretization agent, transformation agent, clean miss agent, clean noisy agent. The responsibilities of each agent include Coordinator agent to act as a manager, who is responsible for coordinating the various tasks that needs to be performed. It determines the required preprocessing task. CleanMiss Agent and clean noisy agent handle missing and noisy data by using various types of techniques based on type of missing and noisy cases. Transformation Agent is used to transform the data into appropriate forms for mining. The role of reduction agent is to discretize the data by using discretization techniques selected.

Data preprocessing application acts as an interface that process the data to be mined. The dataset with inconsistency is stored in the database with the metadata of it. The whole data is analyzed to find the inconsistencies, duplicate fields, multiple data formats in same attribute, missing data field. The aforementioned tasks are performed by the coordinator agent. Based on the operations to be performed it is given to the corresponding agents [16]. The functions of the preprocessing software include data integration, data reduction, data transformation, data cleaning and data visualization. Each of the function is handled by intelligent agents.

Data integration process combines data from multiple sources like data cubes, multiple databases, and flat files. It performs schema integration and also objects matching. It makes use of the metadata for the integration. It performs correlation analysis and also chi-square test to handle redundancy. Data transformation involves smoothing, aggregation, generalization, normalization and also attributes construction. These methods help to make data more appropriate for mining [20].

Strategies in data reduction are data cube aggregation, attribute subset selection, dimensionality reduction, numerosity reduction, and discretization in addition to concept hierarchy generation. This is performed so that the time taken for complex data analysis and also mining huge amounts of data can be avoided [16].

Data cleaning is a two step process which includes handling missing values and handling noisy data. It is accomplished with discrepancy detection that makes use of metadata which provides knowledge about domain and data type. While scanning the dataset, if the tuples have no recorded value then various strategies are used such as ignore the tuple, fill each missing values manually, use the most probable value, use the attribute mean to fill in case of numeric data, use a constant like unknown or infinity. Binning, regression and clustering are carried out to remove the errors and smoothing the data [17].

The whole sequence of operations performed in a dataset is depicted in figure 2.

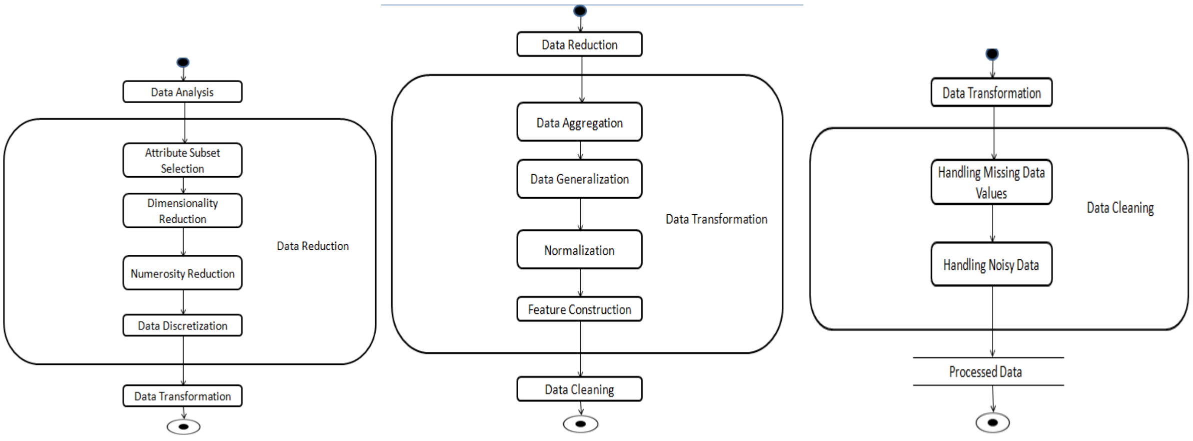


Fig 1: Data Preprocessing Phases

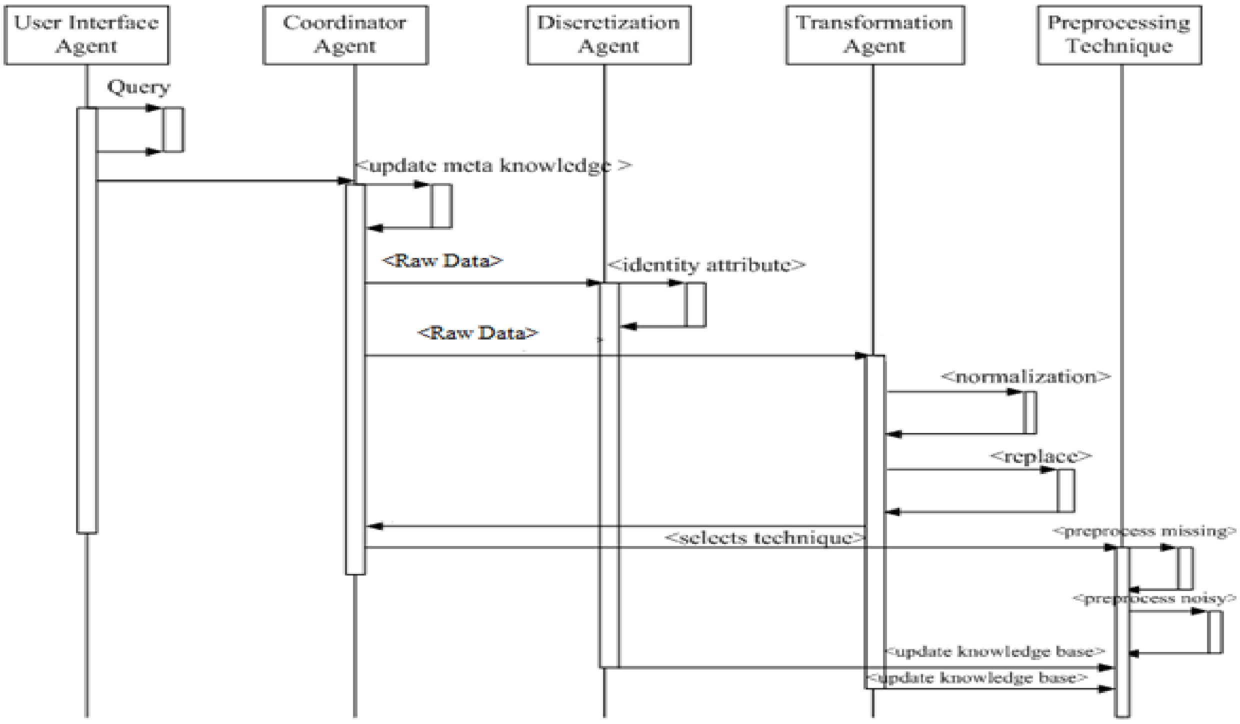


Fig 2: Sequence of process communication

4. IMPLEMENTATION AND RESULT ANALYSIS

In general, data preprocessing is defined as any type of processing that is performed on raw data as preparation for further processing [18]. However, different research fields analyze different kinds of data, and further the methods that fall within the scope of this definition are rather different [19]. This system adopts a pattern recognition approach and refers to preprocessing as tasks that are done after analyzing the data. The techniques to preprocess include cleaning, integration, transformation and reduction. Information cleaning routines works by filling in missing values, smoothing noisy information, identifying or removing outliers, and resolving inconsistencies [15]. The steps of data transformation include normalization where scaling attribute values use mean and standard deviation. This scaling attribute values falls within a specified range. Aggregation step of data transformation move up in the concept hierarchy on numeric attributes and generalization step moves up in the concept hierarchy on nominal attributes [5].

Each phase of the above mentioned preprocessing activity is assigned to an agent to reports ultimately to the coordinator agent. The data base further has the repository of each of these processing activities. The architecture of the preprocessing system is implemented using NetBeans.

The application is uploaded and tested with excel information data file with a maximum size of 10 MB. Table 1 depicts the subset of a sample dataset that is used in testing the implemented data preprocessing system. Once the data is uploaded using the user interface, the file is stored in a specific location. Once the file is uploaded, the inconsistencies are determined by the coordinator agent using association rule and it updates the noisy information. Subsequently, this information will be represented in a file format which will be converted to another format that can be uploaded in the data mining tools available. As an instance, the dataset in excel file format is converted to .arff format suitable for WEKA. All updated data information will consequently be saved in the data repository. Agents will autonomously determine the operations to be performed and executes the same which eventually gets stored in the repository.

Table 1. Subset of sample dataset

Cus_Tno	Mstatus	Po_code	O_code	Cu_em
10	Married	Contractor	Student	9
28	Single	Retired	???	56
536	Divorced	???	Laborer	-198
-627	Widow	Skilled Laborer	Unemployed	???
519	???	Professional	Professional	218
???	Single	Retired	Employ	$$$

4.1 Algorithms

Algorithm for identification of noisy data and analysing of the initial dataset is given further.

Algorithm 1: dataset analysis in the agent based preprocessing system

Input: Set of data files

Output: Preprocessed knowledge information

Step 1: Start

Step 2: Get the data sets C

Step 3: Set of data information parameters

Step 4: L is an empty List (L is a matched concept list n)

Step 5: Sdoc is a new sentence in C

Step 6: Build metadata Cdoc from Sdoc

Step 7: for each concept Ci

Step 8: do

Step 9: Return Cdoc.

Step 10: END for

Step 11: END

Where C = set of data information, doc = documents of data base.

Algorithm 2: Identification of the noisy data

Input: Inconsistent dataset

Output: Processed data.

Step 1: Get the noisy dataset

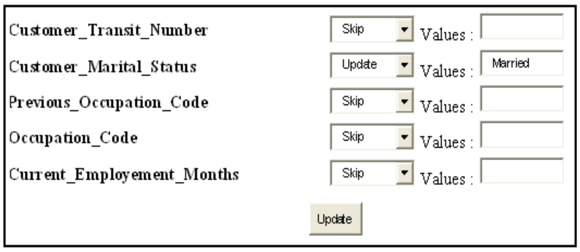
Step 2: Set of the noisy data information parameters

Step 3: Identify the inconsistent knowledge in the dataset using metadata

Step 4: Identify special values and check the data and attributes

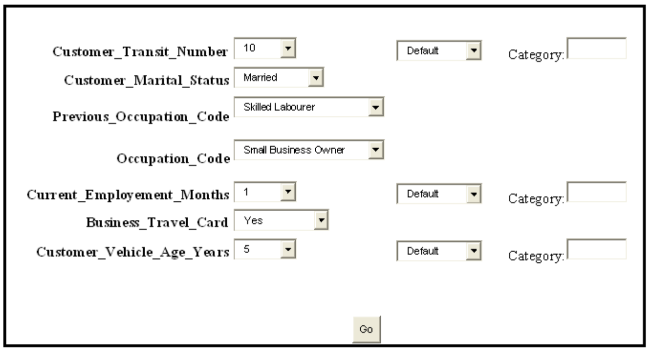
Step 5: Provide the access to data from database for other agents system using coordinator agent.

Step 6: Refine the processed using the user interface agent.



Customer_Transit_Number: Skip, Values :
Customer_Marital_Status: Update, Values : Married
Previous_Occupation_Code: Skip, Values :
Occupation_Code: Skip, Values :
Current_Employement_Months: Skip, Values :

Fig 3: Setting Data Values in Data Cleaning



Customer_Transit_Number: 10, Default, Category:
Customer_Marital_Status: Married
Previous_Occupation_Code: Skilled Labourer
Occupation_Code: Small Business Owner
Current_Employement_Months: 1, Default, Category:
Business_Travel_Card: Yes
Customer_Vehicle_Age_Years: 5, Default, Category:

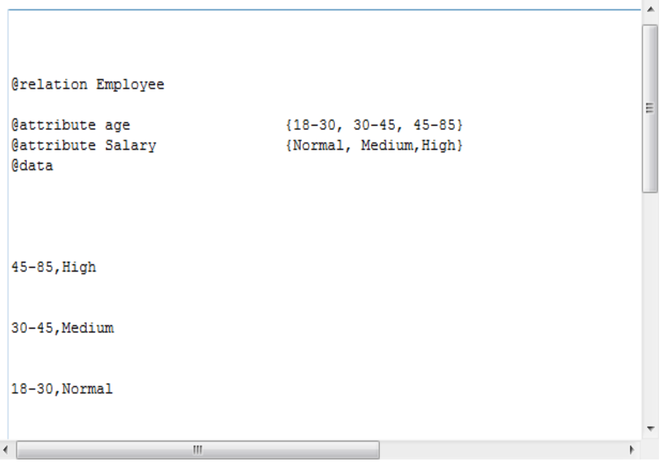
Fig 4: Data Transformation

@attribute Customer_Transit_Number {0-10,10-500,500-1000,1000-4000}

@attribute Customer_Marital_Status {Married,Single,Divorced,Widowed,Separated}
@attribute Previous_Occupation_Code
{Skilled_Labourer,Retired,Small_Business_Owner,Contractor,Business_Executive,Professional,Unemployed,Student,Labourer,Civil_Servant}

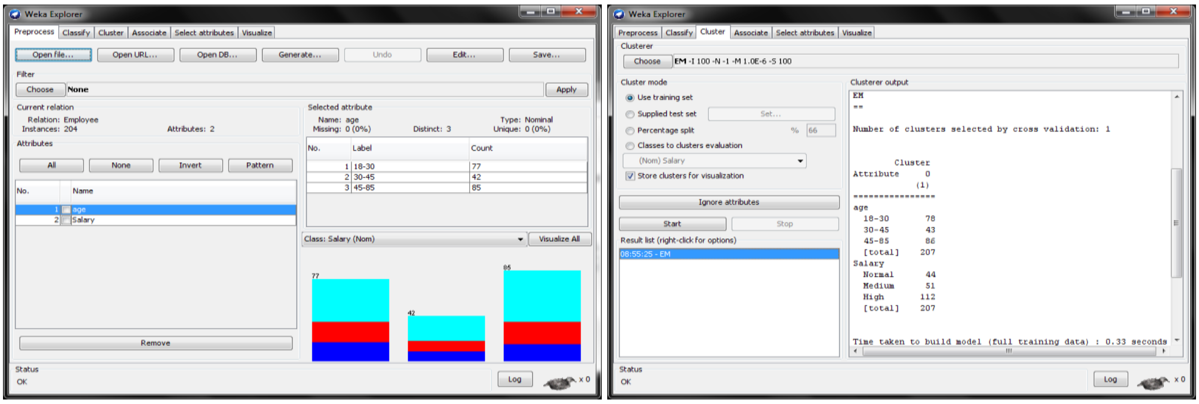
@attribute Occupation Code {Small_Business_Owner,Unemployed,Civil_Servant,Retired,Professional,Contractor,Noisy,Skilled_Trade,Labourer,Executive,Student}
@attribute Current_Employement_Months real
@attribute Bank_Savings_Account {Yes,Other_Bank,Noisy}
@attribute Mastercard_Purchases_Over_last_6_months real
@attribute Loan_Code_if_Applicable {Excellent,Acceptable,Good,Aceptable}
@attribute Department_Store_Credit_Card {Yes,No,Noisy}

Fig 5: Result Processing



```
@relation Employee

@attribute age                  {18-30, 30-45, 45-85}
@attribute Salary               {Normal, Medium,High}
@data



45-85,High


30-45,Medium


18-30,Normal
```

Fig 6: Data Sampling



Filter: Choose None

Current relation
Relation: Employee
Instances: 204 Attributes: 2

Attributes

No.	Name
1	age
2	Salary

Selected attribute
Name: age Type: Nominal
Missing: 0 (0%) Distinct: 3 Unique: 0 (0%)

No.	Label	Count
1	18-30	77
2	30-45	42
3	45-85	85

Class: Salary (Nom)

Status
OK x 0

Clusterer
Choose EM -I 100 -N -1 -M 1.0E-6 -S 100

Cluster mode
- ● Use training set
- ○ Supplied test set
- ○ Percentage split % 66
- ○ Classes to clusters evaluation
- (Nom) Salary
- ☑ Store clusters for visualization

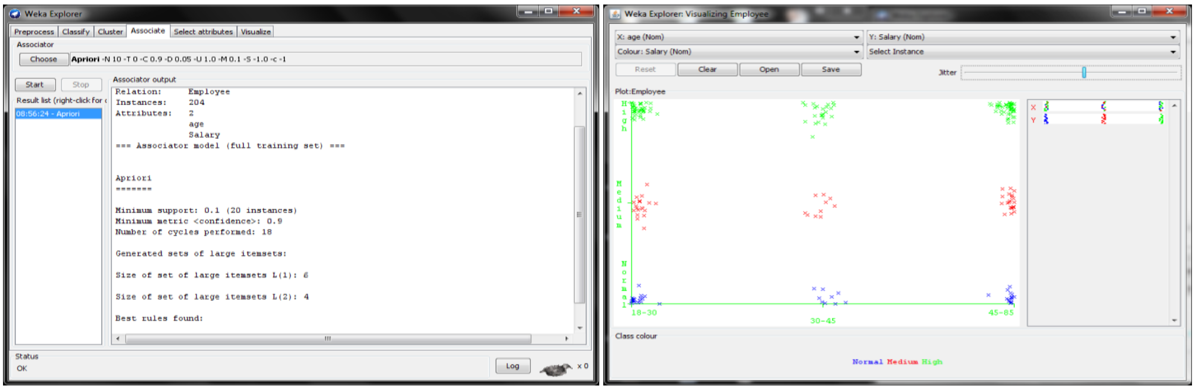
Result list (right-click for options)
08:55:25 - EM

Clusterer output

```
EM
==

Number of clusters selected by cross validation: 1


              Cluster
Attribute           0
                  (1)
================
age
   18-30           78
   30-45           43
   45-85           86
   [total]        207
Salary
   Normal          44
   Medium          51
   High           112
   [total]        207


Time taken to build model (full training data) : 0.33 seconds
```

Status
OK x 0

Fig 7: Analyzing the performance of data preprocessed by uploading in WEKA tool



Associator

Choose: Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

Associator output

```
Relation:     Employee
Instances:    204
Attributes:   2
              age
              Salary
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (20 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 6

Size of set of large itemsets L(2): 4

Best rules found:
```

Status: OK

X: age (Nom)

Y: Salary (Nom)

Colour: Salary (Nom)

Select Instance

Fig 8: Analyzing the performance of data preprocessing by uploading in WEKA tool

5. RESULT ANALYSIS

Figure 3 through Figure 7 depicts the implementation of the data preprocessing system. Figure 3 shows the analysis and result. Figure 4 depicts the data transformation. Figure 5 and Figure 6 shows the result processing and data sampling. Figure 7 and 8 shows the output data analysis in WEKA tool. When compare to the result obtained in WEKA with the same dataset before preprocessing, the result obtained after the data preprocessing is more consistent. Moreover the outliers present in the clustering are less after data preprocessing.

However in this research the system developed is capable of handling data that is of megabyte size and agents are designed only to handle data from one domain. Further the system developed has only a limited set of data preprocessing methods under data reduction and data transformation thereby opening further research on various other data preprocessing techniques that can be incorporated in the system for improved data preparation. Therefore the future work suggested is to apply the system across the wide spectrum of application domain in order to justify and further enhance the capabilities of this newly developed system.

6. CONCLUSION

Advancement in technology enables organizations to store, process and update huge and complex data dynamically. Data mining technique enables one to extract information from the database with the aid of data mining tools and with the intervention of data miner for performing the operations. Since current data mining tools are expensive and inefficient in handling the dynamic data, agents help to resolve the data inconsistencies and reduce the size of data. The characteristics of these agents enable them to adapt in the data preprocessing environment and to analyze the nature of data. The data preprocessing software ensure to improve the quality of data mining as it is designed by considering scalability characteristics. The entire operations are performed intelligently and in automated mode which is more efficient, accurate and less time consuming than when compared with the operations that are performed with human intervention. Intelligent systems thus play an imperative role in developing data preprocessing system that can drastically reduce the complexity of the manual preprocessing activities which is the uniqueness of the system.

7. REFERENCES

[1] Bobek, S. and Perko, I., 2006. Intelligent Agent Based Business Intelligence. University of Maribor, Faculty of Economics and Business, Razlagova ulica14, SI-2000 Maribor, Slovenia.

[2] Han, J. and Kamber, M. 2001. Data Mining: Concepts and Techniques. Simon Fraser University, Academic Press. Information Quality Issues. International Institute for Advanced Studies in Systems Research and Cybernetic, IIAS.

[3] Langseth, J. and Vivarat, N., 2005. Why Proactive Business Intelligence Is A Hallmark Of The Real-Time Enterprise: Outward Bound. Intelligent enterprise.

[4] Lin, L., Osan, R. and Tsien J.Z. 2006. Organizing Principles of Real-time Memory Encoding: Neural Clique Assemblies and Universal Neural Codes. Center for Systems Neurobiology, Departments of Pharmacology and Biomedical Engineering, Boston University, Boston, MA 02118, USA. Shanghai Institute of Brain Functional Genomics, and the Key Laboratory of Chinese Ministry of Education, East China Normal, University, Shanghai 200062, China.

[5] Petteri Nurmi, W.S. 1999. Using Neural Networks for Data Mining. Computer Science Department, Carnegie Mellon University, University of Wisconsin-Madison.

[6] Wook Baik, Mitra, S. Pal, S.K. and Mitra, P. 2002. Data Mining in Soft Computing Framework: A Survey. IEEE Trans. Neural Networks, 13(1):3–14.

[7] Chunsheng Li and Yatian Gao, 2001. An Introduction to Multi agent systems. Department of Computer Science, University of Iiverpool, UK. ISBN 0-471-49691- X.

[8] Nils, N., 1998. Artificial Intelligence: A New Synthesis. Morgan Kaufmann Publishers. ISBN 978-1-55860-467-4.

[9] Padghan, L. and Winikopff, M., 2004. Developing Intelligent Agent Systems. Wiley.

[10] Roya, A., Norwati, M., Nasir, S. (2009). Training Process Reduction Based On Potential Weights Linear Analysis to Accelerate Back Propagation Network, Accepted by International Journal of Computer Science and Information Security (IJCSIS), Vol. 3, No. 1, July

[11] Roya, A., Norwati, M., Nasir, S. and Nematollah S. (2009). New Supervised Multi layer Feed Forward Neural Network model to accelerate classification with high accuracy, Accepted by European Journal of Scientific Research (EJSR), ISSN 1450-216X, Vol. 33 Issue 1, pp.163-178.

[12] Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques,Elsevier, 22-Jun-2011 - 744 pages

[13] Stuart J. R. and Peter N., 2003. Artificial Intelligence: A Modern Approach. Second edition, Upper Saddle River, NJ: Prentice Hall, ISBN 0-13-790395-2.

[14] Sharon Christa, Lakshmi Madhuri, Suma. V, 2012 Data Preprocessing Using Intelligent Agents, International conference on Frontiers in Intelligent Computing Theory and Application, December, Orissa.

[15] Sharon Christa, Suma. V, Lakshmi Madhuri, 2012 An Effective Data Preprocessing Technique for Improved Data Management in a Distributed Environment, Third International Conference on Advanced Computing and Communication Technologies for High Performance Applications- June, Cochin.

[16] T. R. G. Nair, Sharon Christa, Lakshmi Madhuri, Suma. V "Data Preprocessing Model Using Intelligent Agents", International Conference on Information Systems Design and Intelligent Applications- January, 2012 Visakhapatnam.

[17] Vijayan S. 2006.Application of agents and Intelligent Information Technologies. ISBN 1-59904-265-7.Published in USA by Idea group publishing.

[18] Werbos, P.J. 1974. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD Thesis, Harvard University, Cambridge, MA.

[19] Wooldridge, M., 2002. An Introduction to Multi Agent Systems. West Sussex, Willey.

[20] Yoav Sh. and Kevin L., 2009. Multi Agent Systems: Algorithmic, Game- Theoretic, and Logical Foundation. First published. Printed in the USA. ISBN 978-0-521-89943-7.