International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-3 Issue-10, March 2014

Design of an Efficient Clustering Using GNG and SOM

Sundeep Kumar, Shilpi Gupta

Abstract-Clustering is the process of grouping the data into classes so that objects have the high similarity in comparison to one another object within a cluster. Because they are very dissimilar to object in other clusters. Dissimilarities are assessed based on the attribute value describing the object. Different types of raw data are available on the World Wide Web. Various data mining techniques can be applied on raw data to manage and organize like data preprocessing. The preprocess data is achieved through data cleaning, data reduction and data integration algorithm which can be used in variety of applications such as Clustering, Neural Network, association rules, and sequential pattern etc. In this paper we performed the data preprocessing activities like data cleaning, data reduction, data integration and related algorithm. A novel approach Growing Neural Gas and Self Organizing Maps algorithms is introduced and apply on preprocess data for clustering and performance evaluated through certain parameter error graph, time elapsed and mean weight difference kind of clustering.

Keywords: Clustering, Growing Neural Gas (GNG), SOM (Self Organizing Map), Data Preprocessing.

I. INTRODUCTION

Web has recently become a powerful platform for retrieving large amount of raw data that is now in general freely available for user access. Several data preprocessing techniques can be applied for filter the raw data. Preprocessing is necessary, because raw data coming from the web server is noisy, incomplete, duplicacy and having unnecessary symbol. On preprocessed data different data mining techniques [7] can be applied like statistical analysis, association rules, sequential patterns, neural network and clustering. Clustering basically deals with the organization of a set of objects in a multidimensional space into cohesive groups, called clusters. The basic idea is that if a rule is valid for one object, it is very possible that the rule also applies to all the objects that are very similar to it. These objects are placed in the same cluster. Hence it is a form of unsupervised classification, which means that the categories into which the collection must be partitioned are not known. Main objective of this paper is to understand the preprocessing of usage data. On the preprocess data different data mining algorithm can be applied like GNG and SOM algorithm for clustering and evaluate the performance of these algorithm GNG and SOM through certain parameter error graph, time elapsed and mean weight difference kind of clustering. This paper is organized as follows.

In section II an overview of clustering is given which describes applications of clustering. In section III data preprocessing activities like data cleaning, data reduction, and data integration related algorithms are presented. In section IV we discuss proposed approach and Evaluate the Performance of GNG and SOM algorithm in section V through certain parameter like errors graph, time elapsed graph and mean weight difference graph. Conclusion is given in section VI.

Manuscript published on 30 March 2014.
***Correspondence Author(s)**
Sundeep Kumar, Department of Computer Science, Mahamaya Technical University/ JSS Academy of Technical Education/ Noida, India.
Shilpi Gupta, Department of Computer Science, Mahamaya Technical University/ JSS Academy of Technical Education/ Noida, India.

II. CLUSTERING

The process of grouping a set of physical or abstract object into classes of similar object is called clustering. A cluster is a collection of data object that are similar to one another within the same clustering and are dissimilar to the object in other clusters. Clustering is the process of grouping the data into classes so that objects have the high similarity in comparison to one another object within a cluster. Because are very dissimilar to object in other clusters. Dissimilarities are assessed based on the attribute value describing the object. Clustering has its roots in many areas, including data mining, statistics, biology and machine learning. Clustering can also be used for outlier detection. Clustering has been used in many application including biology, medicine, anthropology, marketing and economics. Clustering applications include plant and animal classification, disease classification, image processing, pattern recognition and document retrieval.

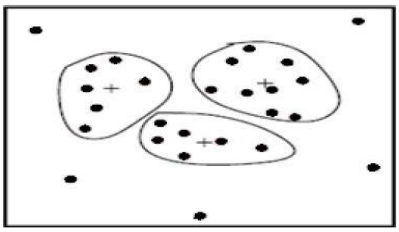

Figure 1: Architecture of Clustering

A. Applications of clustering

i. **Marketing:** Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs.
ii. **Land use:** Identification of areas of similar land use in an earth observation database.
iii. **Insurance:** Identifying groups of motor insurance policy holders with a high average claim cost.
iv. **City-Planning:** Identifying groups of houses according to their house type, value, and geographical location.

III. DATA PREPROCESSING

Data pre-processing is an often neglected but important step in data mining process. Data preprocessing is a data mining technique that involves transforming raw data into an understandable format. Data preprocessing is a proven method of resolving such issues.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Analyzing data that has not been carefully screened for such problem can produce misleading result. It is important to understand that the quality data is a key issue when we are going to mining from it. Nearly 80% of mining efforts often spend to improve the quality of data [2]. The data which is obtained from the web may be incomplete, noisy, duplicate data, having unnecessary symbols and inconsistent. The attributes that we can look for, in quality data includes accuracy, completeness, consistency, timeliness, believability, interpretability and accessibility. Data preprocessing is one of the most critical steps in a data mining process which deals with the preparation of the initial dataset. Raw data is highly susceptible to noise, missing values, duplicacy and inconsistency. The quality of data affects the data mining results. In order to help improve the quality of data and consequently, of the mining results raw data is pre-processed so as to improve the efficiency and ease of the mining process. Data preprocessing is one of the most critical steps in a data mining process which deals with the preparation and transformation of the initial dataset. Data preprocessing methods are dividing into following categories:

- Data Cleaning
- Data Reduction
- Data Integration

A. Data Cleaning

Real-world data tend to be incomplete, noisy, duplicate, having unnecessary symbols and inconsistent. The raw data usually contains a number of unnecessary symbols that can adversely affect the categorization process and do not help in identification of the data. For example, words or special symbols constituting {$ % ^ * ! @ # &} that occur frequently in the data. These symbols need to remove from the data. Even the symbols, such as “” replace with ‘’ and ; is replace with ,(comma) etc. It is important to clean the data in order to increase the efficiency.

Data Cleaning Algorithm is a process used to determine inaccurate, incomplete, or unreasonable data that remove these data from the raw data. The process may include format checks, completeness checks, and reasonable checks, review of the data to identify data [5].

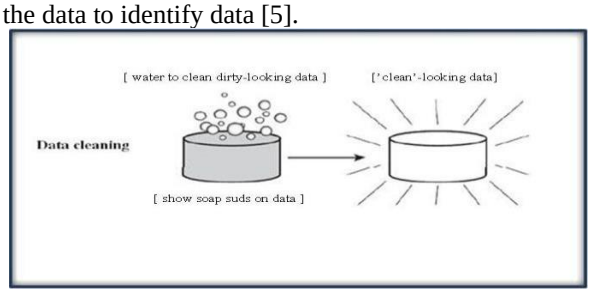

Figure 2: Data Cleaning

Data Cleaning Algorithm

Step 1: Get the file for pre-processing/cleaning using browse button.
Step 2: if upload button is pressed then goto step 3
Step 3: Open and read the file line by line and show the contents on textarea.
Step 4: end if
Step 5: if fields = {$ % ^ * ! @ # &} then goto step 6
Step 6: Remove them and filter the file contents
Step 7: end if
Step 8: if fields= {“” ;} then goto step 9
Step 9: Replace with {‘’ ,}respectively
Step 10: Save the records in database.
Step 11: Read the data from database and display the fields on textarea column-wise.
Step 12: end if

B. Data Reduction

Data reduction techniques have been helpful in analyzing reduced representation of the data set without compromising the integrity of the original data and yet producing the quality of data. Data reduction is a process which is used to remove the duplicate data from the cleaned data. A data usually contains the similar data row that is duplicacy occur in data then remove the duplicacy in the record. For example if one row have that data age = 30, job = unemployed, marital = married, education = primary, balance = 1787 etc. and second row have the same data than remove the duplicate data. The concept of data reduction is commonly understood as either reducing the volume or reducing the dimension.

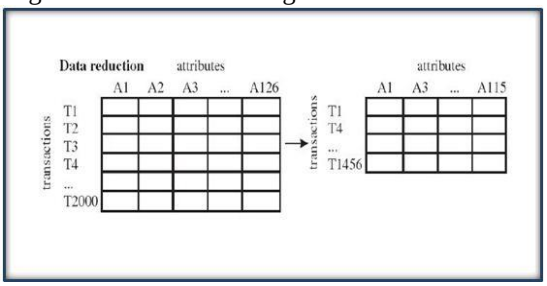

Figure 3: Data Reduction

Data Reducing Algorithm

Step 1: Read the data from database.
Step 2: if data contains duplicate data then goto step 3
Step 3: Remove the duplicate data from the database.
Step 4: end if
Step 5: Read the reduced data from database and display the fields on textarea column-wise.

C. Data Integration

Data Integration is the combination of technical and business processes used to combine data from disparate sources into meaningful and valuable information. Data Integration is a process in which heterogeneous data is retrieved and combined as an incorporated form and structure. It is used to integrate the data from multiple sources. For example applying integrate algorithm on age attribute data is integrated in the ascending order of age. User can integrate all seventeen attributes according to their values. In other way user can arrange the data in different structure like user select the job attribute then all data arrange in a format only for job attribute values according to job type.

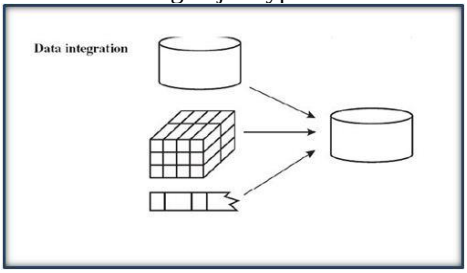

Figure 4: Data Integration Algorithm

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Data Integration Algorithm
Step 1: Read the data from database.
Step 2: if select attribute then goto step 3
Step 3: Display the fields on textarea column-wise in sequence.
Step 4: end if
Step 5: if select attribute == Job then goto step 6
Step 6: Read the data from database according to selected job.
Step 7: Display the data of selected job attribute on textarea column-wise.
Step 8: end if

IV. PROPOSED APPROACHES

In the present work, this paper proposes the data cleaning, data reduction and data integration steps of data preprocessing. These steps are applied on bank raw data. To obtain this result we need to process GNG and SOM algorithm to train the Clustering, and show the comparison between them through certain parameter.

A. Self-Organizing Map (SOM)

The Self-Organizing Map algorithm by Kohonen is a neural network algorithm for creating a topologically correct nonlinear projection of high dimensional data [8] into a neuron lattice of lower dimensionality. The Self-Organizing Map (SOM) algorithm can be used to visualize, cluster and analyze large amounts of multidimensional data in an unsupervised manner [4]. However, in its original form the SOM algorithm lacks the ability to represent temporal information. Several extensions have been developed to add this functionality to the basic SOM algorithm based on delay mechanics, recurrent connections and leaky activation potential.

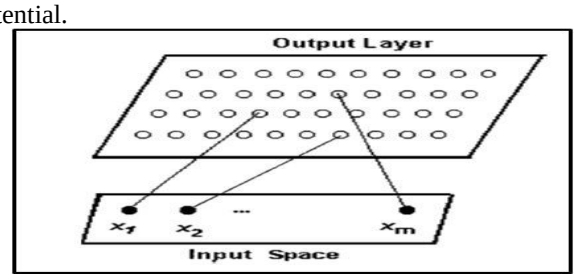

Figure 5: Architecture of Self Organizing Map

Self-Organizing Map (SOM) Algorithm
The Self-Organizing Map algorithm can be broken up into 6 steps [3].
Step 1: Each node's weights are initialized.

Step 2: A vector is chosen at random from the set of training data and presented to the cluster.

Step 3: Every node in the cluster is examined to calculate which ones' weights are most like the input vector. The winning node is commonly known as the Best Matching Unit (BMU).

Calculate the BMU

$$DistFromInput^2 = \sum_{i=0}^{t=n} (I_i - W_i)^2$$

I = current input vector
W = node's weight vector
n = number of weights

Step 4: The radius of the neighborhood of the BMU is calculated. This value starts large. Typically it is set to be the radius of the cluster, diminishing each time-step.

Radius of the neighborhood

$$\sigma(t) = \sigma_0 e^{(-t/\lambda)}$$

t = current iteration
λ = time constant (Equation 2b)
σ_0 = radius of the map

Time Constant
λ = numIteration/mapRadius

Step 5: Any nodes found within the radius of the BMU, calculated in step 4, are adjusted to make them more like the input vector. The closer a node is to the BMU, the more its' weights are altered.

New weight of a Node

$$W(t+1) = W(t) + \Theta(t)\, L(t)(I(t) - W(t))$$

Learning Rate

$$L(t) = L_0\, e^{(-t/\lambda)}$$

Distance from BMU

$$\Theta(t) = e^{(-\text{distFromBMU}^2/(2\sigma^2(t)))}$$

Step 6: Repeat step 2 to step5 for N iterations.

B. Growing Neural Gas (GNG)

Growing Neural gas is an artificial neural network, and introduced in 1991 by Thomas Martinetz and Klaus Schulten. The neural gas is a simple algorithm for finding optimal data representations based on feature vectors. The algorithm was coined "neural gas" because of the dynamics of the feature vectors during the adaptation process, which distribute themselves like a gas within the data space [6]. The GNG algorithm was specifically developed for clustering and vectoraization. It is capable of overcoming some of the major limitations of the standard self-organizing maps. The growth mechanism of the growing cell structures is combined.
Step 1: Create all randomly positioned nodes; with weights Ws i.e. x and y values and plot in the graphs and set their errors to 0.
Step 2: Generate an input vector $\bar{x}$ i.e. Signal X, Signal Y conforming to some distribution.
Step 3: Locate the two nodes *s* and *t* nearest to $\bar{x}$, that is, the two nodes with reference vectors *ws* and wt such that $\| \bar{w}_s - \bar{x} \|^2$ *s* is the smallest value and $\| \bar{w}_t - \bar{x} \|^2$ t is the second smallest, for all nodes *k*.
Step 4: The winner-node *s* must update its local error variable so we add the squared distance between s and $\bar{x}$, to errors.

$$error_s \leftarrow error_s + \| \bar{w}_s - \bar{x} \|^2$$

Step 5: Move *s* and it's topological neighbors towards $\bar{x}$ by fractions *ew* and *en* of the distance $e_w, e_n \in [0,1]$.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

$$\bar{w}_s \leftarrow \bar{w}_s + e_w(\bar{x} - \bar{w}_s)$$

$$\bar{w}_n \leftarrow \bar{w}_n + e_n(\bar{x} - \bar{w}_n), \forall n \in Neighbour(s)$$

Step 6: If the current iteration is an integer multiple of λ and the maximum node counts have not been reached, and then insert a new node. Insertion of a new node *r* is done as follows:

- Find the node *u* with largest error.
- Among the neighbours of *u*, find the node *v* with the largest error.
- Insert the new node *r* between *u* and *v* as follows:

$$\bar{w}_r \leftarrow \frac{(\bar{w}_u + \bar{w}_v)}{2}$$

Step 7: Decrease the error-variables of *u* and *v* and set the error of node *r*.

$$error_u \leftarrow \alpha \times error_u$$

$$error_v \leftarrow \alpha \times error_v$$

$$error_r \leftarrow error_u$$

Step 8: Decrease all error-variables of all nodes j by a factorβ**.**

Step 9: If the stopping criterion is not met then repeat. The criterion might be for example the performance on a test set is good enough, or a maximum number of nodes have been reached, etc.

V.PERFORMANCE EVALUATION

We evaluate the performance of GNG and SOM algorithm for making cluster through these three graphs Error Graph, Time Elapsed Graph and Mean Weight Difference Graph. All the implementation done in java programing language, java servlet used for logical part using Apache Tomcat server 7.0.14.0 and MySQL database used as backend. Netbean IDE 7.3 used to run the project on Windows 7 operating system. These graphs are as follows:

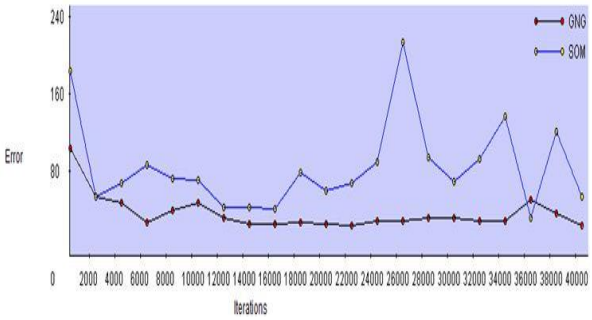

Figure 6: Error Graph

GNG algorithm compute the minimum error as compare to SOM algorithm for making cluster. Figure 6 evaluate the performance of GNG algorithm and SOM algorithm by error graph.

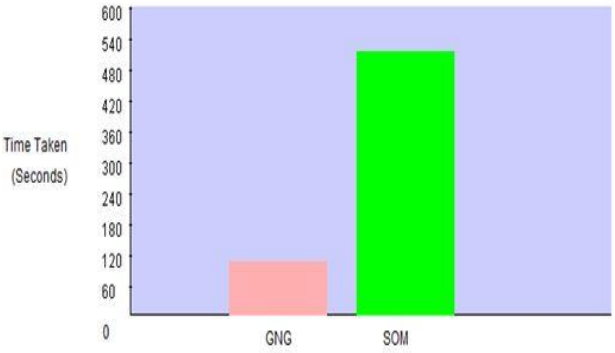

Figure 7: Time Elapsed Graph

GNG algorithm takes the minimum time as compare to SOM algorithm for making cluster. Figure 7 evaluate the performance of GNG algorithm and SOM algorithm by time elapsed graph.

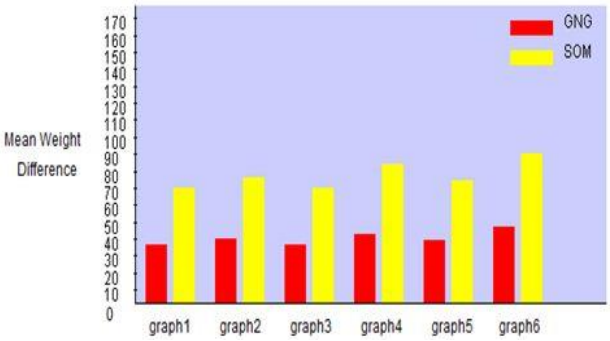

Figure 8: Mean Weight Difference

GNG algorithm takes the mean weight as compare to SOM algorithm for making cluster. Figure 8 evaluate the performance of GNG algorithm and SOM algorithm by mean weight difference graph.

VI.CONCLUSION

This work is done using three parameters to display the result of comparison between GNG algorithm and SOM algorithm. We can conclude that, to identify accurate data in Web to help for clustering by GNG and SOM algorithm. The performance evaluation result obtained under the parameters like errors graph method, time elapsed graph, and mean weight difference graph method. Using performance evaluation, we have shown that GNG algorithm has better for making clustering with comparison to SOM algorithm. In other way GNG build some gathering with great quality of cluster of the preprocessed data. Preprocess data can be used for pattern discovery. GNG can be implemented for documents clustering. The work can be carried out on live media (Web Search Engine). GNG algorithm can also be used for image clustering.

REFERENCES

1. Vaishali A. Zilpe, Dr. Mohammad Atique, 2011. "Neural Network Approach for Web Usage Mining" National Conference on Emerging Trends in Computer Science and Information Technology (ETCSIT).
2. N Tyagi, A. Solanki and S. Tyagi, 2010. "An algorithmic Approach to Data Pre-processing in Web Usage Mining", Int. Journal of Information Technology and Knowledge Management, Volume 2, No. 2, pp. 279-283.
3. Shyam M. Guthikonda, 2005. "Kohonen Self-Organizing Maps", shyamguth ATgmail.com Wittenberg University
4. Melody Y. Kiang, 2001, "Extending the Kohonen Self-Organizing Map Networks for Clustering Analysis" Computational Statistics & Data Analysis 38, pp 161–180.
5. Rajashree Y.Patil, Dr. R.V.Kulkarni, 2012. "A Review of Data Cleaning Algorithms for Data Warehouse Systems", International Journal of Computer Science and Information Technologies, Vol. 3(5), 5212 – 5214.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

6. Anshuman Sharma, 2011. "Web Usage Mining Using Neural Network", International Journal of Reviews in Computing 10th April 2012. Vol. 9.
7. http://www.w3.org/Daemon/user/config/logging.html common - log – file -format.
8. HUILIN YE, BRUCE W.N. LO, 2000. "Feature Competitive Algorithm for Dimension Reduction of the Self-Organizing Map Input Space", Kluwer Academic Publishers, Manufactured in the Netherlands, Applied Intelligence 13, 215–230.

AUTHOR PROFILE

Sundeep Kumar received his MCA Degree in Computer Applications from Gautam Budha Technical University, Lucknow, India in year 2011 and pursuing his M.Tech degree in Computer Science & Engineering from JSS Academy of Technical Education Noida, Uttar Pradesh, India. His area of interest is in Data Mining and Clustering.

Shilpi Gupta received her M.Tech degree in Computer Science from Maharshi Dayanand University, Rohtak, Haryana, India in year 2012 and B.Tech. Degree in Computer Science and Engineering from Uttar Pradesh Technical University, Lucknow, Uttar Pradesh, India in year 2005. She is working as an Assistant Professor in the Department of Computer Science and Engineering at JSS Academy of Technical Education, Noida, India and have more than six year experience in academic.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)