Performance Improvement Of Web Usage Mining By Using Learning Based K-Mean Clustering

Ms. Vinita Shrivastava
M.Tech (Information Technology)
Technocrats Institute of Technology
Bhopal , India
vinitashri24@gmail.com

Mr. Neetesh Gupta
Head Of Department (Information technology)
Technocrats Institute of Technology
Bhopal, India
gupta_neetesh81@yahoo.co.in

Abstract **- Due to the increasing amount of data available online, the World Wide Web has becoming one of the most valuable resources for information retrievals and knowledge discoveries. Web mining technologies are the right solutions for knowledge discovery on the Web. The knowledge extracted from the Web can be used to raise the performances for Web information retrievals, question answering, and Web based data warehousing. In the present work, we propose a new technique to enhance the learning capabilities and reduce the computation intensity of a competitive learning multi-layered neural network using the K-means clustering algorithm. The proposed model use multi-layered network architecture with a back propagation learning mechanism to discover and analyze useful knowledge from the available Web log data.**

Keywords **- Clustering algorithms, data mining, Unsupervised Learning algorithm,k-mean clustering, web usage mining.**

I. INTRODUCTION

Web mining methodologies can generally be classified into one of three distinct categories: web usage mining, web structure mining, and web content mining examine web page usage patterns in order to learn about a web system's users or the relationships between the documents. In web usage mining the goal is to examine web page usage patterns in order to learn about a web system's users or the relationships between the documents. For example, the tool presented and creates association rules from web access logs, which store the identity of pages accessed by users along with other information such as when the pages were accessed and by whom; these logs are the focus of the data mining effort, rather than the actual web pages themselves.

Data mining is a set of techniques and tools used to the no trivial process of extracting and present implicit knowledge, no knowledge before, this information is useful and human reliable; this is processing from a great set of data; with the object of describing in automatic way models, no knowledge before; to detect tendencies and patterns [1,2]. The Web Usage Mining is the process of applying techniques to detect patterns of usage to Web Page [3,4]. The Web Usage Mining use the data storage in the Log files of Web server as first resource; in this file the Web server register the access at each resource in the server by the users [5,6].

II. NEURAL NETWORK

An Artificial Neural Network (ANN) is an information-processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process [7].

A. Architecture of neural networks

1) Feed-forward networks

Feed-forward ANNs allow signals to travel one way only; from input to output. There is no feedback (loops) i.e. the output of any layer does not affect that same layer. Feed-forward ANNs tend to be straightforward networks that associate inputs with outputs. They are extensively used in pattern recognition. This type of organization is also referred to as bottom-up or top-down.

2) Feedback networks

Feedback networks can have signals traveling in both directions by introducing loops in the network. Feedback networks are very powerful and can get extremely complicated. Feedback networks are dynamic; their 'state' is changing continuously until they reach an equilibrium point. They remain at the equilibrium point until the input changes and a new equilibrium needs to be found. Feedback architectures are also referred to as interactive or recurrent,

although the latter term is often used to denote feedback connections in single-layer organizations.

III. MINING WEB USAGE DATA

The information provided by the data sources described above can be used to construct several data abstractions, namely users, page-views, click-streams, and server sessions. A user is defined as a single individual that is accessing file web servers through a browser. In practice, it is very difficult to uniquely and repeatedly identify users. A page-view consists of every file that contributes to the display on a user's browser at one time and is usually associated with a single user action such as a mouse-click. A click-stream is a sequential series of page-views requests. A server session (or visit) is the click-stream for a single user for a particular Web site. The end of a server session is defined as the point when the user's browsing session at that site has ended [3, 8]. The process of Web usage mining can be divided into three phases: preprocessing, pattern discovery, and pattern analysis [3, 9].

Preprocessing consists of converting usage information contained in the various available data sources into the data abstractions necessary for pattern discovery. Another task is the treatment of outliers, errors, and incomplete data that can easily occur due reasons inherent to web browsing. The data recorded in server logs reflects the (possibly concurrent) access of a Web site by multiple users, and only the IP address, agent, and server side click-stream are available to Identify users and server sessions. The Web server can also store other kinds of usage information such as cookies, which are markers generated by the Web server for individual client browsers to automatically track the site visitors [3, 4]. After each user has been identified (through cookies, logins, or IP/agent analysis), the click-stream for each user must be divided into sessions. As we cannot know when the user has left the Web site, a timeout is often used as the default method of breaking a user's click-stream into sessions [2].

The next phase is the pattern discovery phase. Methods and algorithms used in this phase have been developed from several fields such as statistics, machine learning, and databases. This phase of Web usage mining has three main operations of interest: association (i.e. which pages tend to be accessed together), clustering (i.e. finding groups of users, transactions, pages, etc.), and sequential analysis (the order in which web pages tend to be accessed) [3, 5]. The first two are the focus of our ongoing work. Pattern analysis is the last phase in the overall process of Web usage mining. In this phase the motivation is to filter out uninteresting rules or patterns found in the previous phase. Visualization techniques are useful to help application domains expert analyze the discovered patterns.

IV. CONVENTIONAL METHOD USED IN WEB MINING

A. Clustering

Clustering the process of partition a set of data in a set of meaning full subclasses known as clusters. It helps users understand the natural grouping or structure in a data set. Clustering is an unsupervised learning technique which aim is to find structure in a collection of unlabeled data. It is being used in many fields such as data mining, knowledge discovery, pattern recognition and classification [3].

Central clustering algorithms [4] are often more efficient than similarity-based clustering algorithms. We choose centroid-based clustering over similarity-based clustering. We could not efficiently get a desired number of clusters, e.g., 100 as set by users. Similarity-based algorithms usually have a complexity of at least O (N2) (for computing the data-pair wise proximity measures), where N is the number of data instances.

In contrast, centroid-based algorithms are more scalable, with a complexity of O (NKM), where K is the number of clusters and M the number of batch iterations. In addition, all these centroid-based clustering techniques have an online version, which can be suitably used for adaptive attack detection in a data environment

B. K-Mean Algorithm

The K-Means algorithm is one of a group of algorithms called partitioning clustering algorithm [4]. The most commonly use partitional clustering strategy is based on square error criterion. The general objective is to obtain the partition that, for a fixed number of clusters, minimizes the total square errors.

Suppose that the given set of N samples in an n-dimensional space has somehow been partitioned into K-clusters $\{C_1, C_2, C_3 \ldots C_K\}$. Each C_K has n_K samples and each sample is in exactly one cluster, so that $\sum n_K = N$, where k=1… K. The mean vector M_k of cluster C_K is defined as the centroid of the cluster

$$M_K = (1/n_k) \sum_{i=1}^{n_k} x_{ik} \quad (1)$$

Where x_{ik} is the i^{th} sample belonging to cluster C_K. The square-error for cluster C_K is the sum of the squared Euclidean distances between each sample in C_K and its centroid. This error is also called the within-cluster variation [5]:

$$e_k^2 = \sum_{i=1}^{n_k} (x_{ik} - M_k)^2 \quad (2)$$

The square-error for the entire clustering space containing K cluster is the sum of the within-cluster variations

$$E_k^2 = \sum_{k=1}^{K} e_k^2 \quad (3)$$

The basic steps of the K-mean algorithm are:

- Select an initial partition with K clusters containing randomly chosen sample, and compute the centroids of the clusters.
- Generate a new partition by assigning each sample to the closest cluster center.
- Compute new cluster centre as the centroids of the clusters.
- Repeat steps 2 and 3 until optimum value of the criterion function is found or until the cluster membership stabilizes.

V. PROPOSED APPROACH

In the present work, the role of the k-means algorithm is to reduce the computation intensity of the neural network, by reducing the input set of samples to be learned. This can be achieved by clustering the input dataset using the k-means algorithm, and then take only discriminate samples from the resulting clustering schema to perform the learning process.

The number of fixed clusters can be varied to specify the coverage repartition of the samples. The number of selected samples for each class is also a parameter of the selection algorithm. Then, for each class, we specify the number of samples to be selected according to the class size. When the clustering is achieved, samples are taken from the different obtained clusters according to their relative intraclass variance and their density. The two measurements are combined to compute a coverage factor for each cluster. The number of samples taken from a given cluster is proportional to the computed coverage factor. Let A be a given class, to witch we want to apply the proposed approach to extract S sample. Let k be the number of cluster fixed to be used during the k-means clustering phase. For each generated cluster cli, (i:1..k), the relative variance is computed using the following expression:

$$Vr(cl_i) = \frac{\frac{1}{Card(cl_i)} * \sum_{x \in cl_i} dist(x, c_i)}{\sum_{j=1}^{k} \left(\frac{1}{Card(cl_j)} * \sum_{x \in A} dist(x, c_j) \right)} \quad (4)$$

When Card(X) give the cardinality of a given set X, and dist(x,y) give the distance between the two points x and y.

The most commonly used distance measure is the Euclidean metric which defines the distance between two points x=(p1,….pN) and y=(q1,….,qN) from RN as:

$$dist(x, y) = \sqrt{\sum_{i=1}^{N} (p_i - q_i)^2} \quad (5)$$

The density value corresponding to the same cluster cli is computed like the following:

$$Den(cl_i) = \frac{Card(cl_i)}{Card(A)} \quad (6)$$

The coverage factor is then computed by:

$$Cov(cl_i) = \frac{(Vr(cl_i) + Den(cl_i))}{2} \quad (7)$$

We can clearly see that: $0 \leq Vr(cli) \leq 1$ and $0 \leq Den(cli) \leq 1$ for any cluster cli. So the coverage factor Cov(cli) belong also to 1-Cluster the class A using the k-means algorithm into k cluster.the [0,1] interval. Furthermore, it is clear that:

$$\sum_{i=1}^{k} Vr(cl_i) = 1 \quad \text{and} \quad \sum_{i=1}^{k} Den(cl_i) = 1$$

We can so deduce easily that:

$$\sum_{i=1}^{k} Cov(cl_i) = 1$$

Hence, the number of samples selected from each cluster is determined using the expression

Num_samples(cli)=Round(S*cov(cli)

Let A be the input class; k: the number of cluster; S: the number of samples to be selected ($S \geq k$); Sam(i): the resulting selected set of samples for the cluster i; Out_sam: the output set of samples

selected from the class A; Candidates: a temporary array that contain the cluster points and their respective distance from the centroid. i,j,min,x: intermediates variables; ε: Neiberhood parameter

The proposed selection model algorithm is

- Cluster the class A using the k-means algorithm into k cluster.
- For each cluster cli (i:1..k) do

```
{  Sam(i) :={centroid(cli)};
        j:=1;

For each x from cli do
{ Candidates [j].point :=x;
Candidates [j].location :=dist(x, centroid(cli)) ;
j:=j+1 ;};
```

Sort the array Candidates in descending order with Hence, the number of samples selected from each cluster is respect to the values of location field;

```
  j:=1;
While((card(Sam(i)))<Num_samples(cli))
and (j<card(cli)) do{min:=100000;
For each x from Sam(i) do
  {if dist(Candidates[j].point,x)<min
        then min:= dist(Candidates[j].point,x) ;
          }
         if (min > ε) then
      Sam(i):=Sam(i) U{Candidates[j].point};
j:=j+1; }
     if card(Sam(i)) < Num_samples(cli) then
repeat{Sam(i):=Sam(i)UCandidates[random].poin
}until (card(Sam(i)) = Num_samples(cli));
```

- For i=1 to k do Out_sam:=Out_sam U Sam(i);

VI. RESULT ANALYSIS

A. Data Set Description

Web data clustering is the process of grouping Web data into "clusters" so that similar objects are in the same class and dissimilar objects are in different classes [2, 7]. Its goal is to organize data circulated over the Web into groups / collections in order to facilitate data availability and accessing.

We can broadly categorize Web data clustering into (i) users' sessions-based [3,6,10,11,12] and (ii) link-based. [11,13,14] The former uses the Web log data and tries to group together a set of users' navigation sessions having similar characteristics. In this framework, Web-log data provide information about activities performed by a user from the moment the user enters a Web site to the moment the same user leaves it [9]. The records of users' actions within a Web site are stored in a log file. Each record in the log file contains the client's IP address, the date and time the request is received, the requested object and some additional information -such as protocol of request, size of the object etc. Figure 1 presents a sample of a Web access log file from a Web server.

```
141.243.1.172 [29:23:53:25] "GET /Software.html HTTP/1.0" 200 1497
query2.lycos.cs.cmu.edu [29:23:53:36] "GET /Consumer.html HTTP/1.0" 200 1325
tanuki.twics.com [29:23:53:53] "GET /News.html HTTP/1.0" 200 1014
wpbfl2-45.gate.net [29:23:54:15] "GET / HTTP/1.0" 200 4889
wpbfl2-45.gate.net [29:23:54:16] "GET /icons/circle_logo_small.gif HTTP/1.0" 200
2624
wpbfl2-45.gate.net [29:23:54:18] "GET /logos/small_gopher.gif HTTP/1.0" 200 935
```

Figure1: A sample of Web Server Log File

B. Data Preprocessing

we need to do some data processing, such as invalid data cleaning and session identification [9]. Data cleaning removes log entries (e.g. images, java scripts etc) that are not needed for the mining process. In order to identify unique users' sessions, heuristic methods are (mainly) used [9], based on IP and session time-outs. In this context, it is considered that a new session is created when a new IP address is encountered or if the visiting page time exceeds a time threshold (e.g. 30 minutes) for the same IP-address. Then, the original Web logs are transferred into user access session datasets for analysis.

C. Web URL

Resources in the World Wide Web are uniformly identified by means of URLs (Uniform Resource Locators). The syntax of an http URL is: '

http://' host.domain [':'port] [abs path ['?' query]]

Where {host.domain [: port] is the name of the server site. The TCP/IP port is optional (the default port is 80),{ abs path is the absolute path of the requested resource in the server file system. We further consider abs path of the form path '/' filename ['.' extension], i.e. consisting of the file system path, filename and ¯le extension. {Query is an optional collection of parameters, to be passed as an input to a resource that is actually an executable program, e.g. a CGI script. On the one side, there are a number of normalizations that must be performed on URLs, in order to remove irrelevant syntactic differences (e.g., the host can be in IP format or host format 131.114.2.91 is the same host as kdd.di.unipi.it).

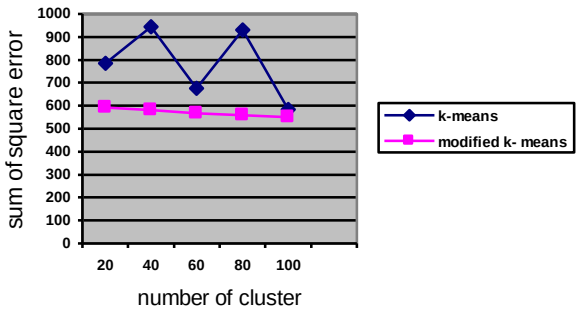
D. Empirical Setting

The K-Means and Modified K-Mean algorithms are written in visual basic 6.0 as front-end and MS-Access used as Backend and compiled into mix files. K-Mean algorithms are relatively efficient due to vectored programming and active optimization. All experiments are run on a PC with a 3.06GHz Pentium-4 CPU with 1GB DRAM and running Windows XP. For the Modified K-Mean Algorithm, the learning rate follows m = 0.5.

TABLE I. A COMPARISON BETWEEN K-MEAN AND MODIFIED K-MEAN ALGORITHM

Cluster	K-Mean Algorithm (SSE)	Modified K-Mean Algorithm (SSE)
20	784	593
40	943	581
60	677	568
80	930	558
100	584	549

graphical representation of the above table



As it is seen from the table and its graph, that when the no. of clusters changes SSE for K mean also changes but in modified algorithm the values of SSE is slow decreased compared to k-mean algorithm. As a summary, we can easily say that our new Modified K-Mean algorithm performs much better than the K-Mean algorithm in discovering user sessions for all kinds of parameters.

VII. CONCLUSION

In this work, we study the possible use of the neural networks learning capabilities to classify the web traffic data mining set. The discovery of useful knowledge, user information and server access patterns allows Web based organizations to mining user access patterns and helps in future developments, maintenance planning and also to target more rigorous advertising campaigns aimed at groups of users. Previous studies have indicated that the size of the Website and its traffic often imposes a serious constraint on the scalability of the methods. As popularity of the web continues to increase, there is a growing need to develop tools and techniques that will help improve its overall usefulness.

REFERENCES

[1] W.J. Frawley, G. Piatetsky-Shapiro, and C.J. Matheus, "Knowledge Discovery in Databases: An Overview", Knowledge Discovery in Databases, G. Piatetsky-Shapiro and W.J Frawley, eds., Cambridge, Mass.: AAAI/MIT Press, pp. 1-27, 1991.

[2] Mika Klemettinen, Heikki Mannila, Hannu Toivonen: A Data Mining Methodology and Its Application to Semi-automatic Knowledge Acquisition. DEXA Workshop 1997: 670-677

[3] R. Kosala, H. Blockeel, and Web Mining Research: A Survey, SIGKKD Explorations, vol. 2(1), July 2000.

[4] J. Srivastava, R. Cooley, M. Deshpande, P.-N. Tan, Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data, SIGKKD Explorations, vol.1, Jan 2000.

[5] Borges-Levene, "An average linear time algorithm for web usage mining:", 2000

[6] P. Batista, M. J. Silva, "Mining web access logs of an on-line newspaper," (2002), http://www.ectrl.itc.it/rpec/RPEC-apers/11-batista.pdf.

[7] Chau, M.; Chen, H., "Incorporating Web Analysis Into Neural Networks: An Example in Hopfield Net Searching", IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, Volume 37, Issue 3, May 2007 Page(s):352 – 358

[8] Raju, G.T.; Satyanarayana, P. S. "Knowledge Discovery from Web Usage Data: Extraction of Sequential Patterns through ART1 Neural Network Based Clustering Algorithm", International Conference on Computational Intelligence and Multimedia Applications, 2007, Volume 2, Issue , 13-15 Dec. 2007 Pages :88 -92

[9] R. Iváncsy, I. Vajk, Different Aspects of Web Log Mining. 6th International Symposium of Hungarian Researchers on Computational Intelligence. Budapest, Nov., 2005.

[10] S. Deerwester, S. Dumains, G. Furnas, T. Landauer and R. Harshman (1990). Indexing by Latent Symantic Analysis. Journal of American Society for Information Science. 41(6): 391-407.

[11] J. Borges, M. Levene (1998), "Mining Association Rules in Hypertext Databases", in Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98), New York City.

[12] M. Spiliopoulou, L.C. Faulstich (1998). WUM: A tool for Web Utilization analysis. Proceedings EDBT workshop WebDB'98, LNCS 1590, Springer, Berlin, Germany. 184-203

[13] A. G. Buchner, & M. D. Mulvenna (1998). Discovering Internet marketing intelligence through online analytical web usage mining. SIGMOD Record, 27 (4), 54-61.

[14] O. Etzioni (1996), "The World Wide Web: Quagmire or Gold Mine", in Communications of the ACM, 39(11):65-68.