ISSN:2249-5789
Christy V et al , International Journal of Computer Science & Communication Networks,Vol 3(3),168-172

Advanced Web Usage Mining Algorithm using Neural Network and Principal Component Analysis

Arumugam, P. and Christy, V

Department of Statistics, Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India.

Abstract

Web Usage Mining becomes a vital aspect in network traffic analysis. Previous study on Web usage mining using a synchronized Clustering, Neural based approach has shown that the usage trend analysis very much depends on the performance of the clustering of the number of requests. Self Organizing Networks is useful for representation of building unsupervised learning, clustering, and Visualization and feature maps. The preprocessed web log files are used for clustering. Growing Neural Gas is one of the types of Self Organizing Networks. The process details the transformation necessaries to adapt the data storage in the Web Servers Log files to an input of Growing Neural Gas algorithm so that we get the result without supervising the trained network. In this paper we are presenting a novel algorithm for clustering the web usage mining data to detect patterns. Self Organizing Map identifies the Winning neurons which are used in growing neural gas algorithm with Euclidean distance measure. Thus the proposed algorithm is hybrid and it combines Artificial Neural Network (ANN) and Principal Component Analysis (PCA).

Keywords-*Growing Neural Gas, Clustering, PCA (Principal Component Analysis), Artificial Neural Network (ANN), Web Log Clustering Algorithm, Self Organizing Map (SOM).*

1. Introduction

Web Mining has been the focus of modern research projects. Artificial Neural Networks (ANNs) resemble an influential implementation to separate clusters of feature vectors in a high dimensional space. The goal of clustering is grouping of data into similar objects. Clustering Analysis is an unsupervised method of partitioning the data set into subsets of similar data objects. Most traditional unsupervised clustering algorithms, such as k-means and fuzzy c-means, partition the data set based on similarity measurement of data. However, one might want to group similar objects with the same class labels when these labels are known. Clustering can be done by a supervised clustering algorithm. Supervised clustering uses extra information, usually represented by labels of objects, to guide the partitioning of data objects into optimal clusters. Neural gas algorithm is a clustering algorithm, which allows you to find a generalization for a set of data represented as a group with similar characteristics. The task of clustering can be realized in many ways, for example: finding the number of groups, clustering a set of n-groups. Each group is a vector of features. Other examples of such algorithms are "k-means method," "self-organizing maps", and its own dynamic modification "growing neural gas" We are combining SOM, GNG for our new approach of clustering for web logs.

2. Related Works

Recently, supervised clustering algorithms have been used in many applications for data mining [1] and bioinformatics [2]. Several supervised clustering algorithms have been proposed. Slonim and Tishby [3] proposed a bottom up agglomerative algorithm, based on hierarchical approaches. Aguilar [4] also proposed a bottom-up agglomerative algorithm by merging neighboring clusters labeled with the same class. Zeidat [1] introduced three clustering algorithms, namely *SPAM* (a variant of the Partitioning Around Medios (*PAM*) algorithm), *SRIDHCR* (which uses a random search and greedy approach for selecting a data sample to be a representative of a cluster), and *SCEC* (which uses a evolutionary computation to seek the optimal set of representatives). The above three algorithms helps to minimize the fitness function that measures the class impurity against the number of clusters. Pedrycz and Vukovich [5] proposed the supervision Algorithm based on fuzzy c-means. The algorithm includes a class constraint factor in the objective function of the original fuzzy c-means algorithm. The drawbacks of the algorithm is that it can handle only a two-class problem, and that the number of clusters must be predefined. A. Jirayusakul and S. Auwatanamongkol [6] proposed optimal guaranteed cluster algorithm is the Supervised Growing neural gas Algorithm. Some algorithms require a predefined number of optimal

clusters, which may not be attainable. The most popular method to clustering analysis is the self organizing feature map (SOM) proposed by Kohonen [9]. The alternative method of clustering is Neural Gas algorithm, Martinatz at al.,[10], it has been applied to vector quantization, prediction, topology representation etc. James and Janusz [11] proposed Self Organizing Neural Gas (SONG), here the error is calculated after the node position updates rather than before. Therefore, we propose a novel hybrid algorithm, hybrid in the sense it utilizes several techniques from supervised and unsupervised learning methods.

In this paper, we combine the self organizing neural Gas with principal component analysis, namely SONG-PCA. The Algorithm is different and it applies some basic Euclidean distance measure with ANN.

.

3. Introduction to Artificial Neural Network

An Artificial Neural Network (ANN) is an information-processing paradigm that has been stimulated by the way of biological nervous systems, such as the brain, process information. The key element of ANN 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 in specific applications like pattern recognition or data classification, through learning process Neural networks, with their incredible ability to develop meaning from complex or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. ANN has Adaptive learning, It is an ability to learn how to do tasks based on the data given for training or initial experience. Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and Manufactured which take advantage of this capability. Fault Tolerance via Redundant Information Coding: Partial destruction of a network leads to the corresponding degradation of redundant information. Also we do have Self-Organization, create its own organization or representation of the information it receives during learning time. There are some method for this are called as SOM (self organization methods).

Self-organizing feature maps (SOFM) is to classify input vectors according to how they are grouped in the input space. They differ from competitive layers in that neighboring neurons in the self-organizing map learn to recognize neighboring sections of the input space. Thus, self-organizing maps learn both the distribution (as do competitive layers) and topology of the input vectors they are trained on. The neurons in the layer of an SOFM are arranged originally in physical positions according to a topology function.

Here a self-organizing feature map network identifies a winning neuron i^* using the same procedure as employed by a competitive layer. However, instead of updating only the winning neuron, all neurons within a certain neighborhood $N_{i^*}(d)$ of the winning neuron are updated, using the Kohonen rule. Specifically, all such neurons $i \in N_{i^*}(d)$ are adjusted as follows:

$$i^{w(q)} = i^{w(q-1)+\propto(p(q)} - i^{w(q-1)} \text{ or}$$

$$i^{w(q)} = (1-\propto) i^{w(q-1)} + \propto p(q)$$

Here the *neighborhood* $N_{i^*}(d)$ contains the indices for all of the neurons that lie within a radius d of the winning neuron i^*.

$$N_i(d) = \{j, d_{ij} \leq \mathrm{d}\}$$

Thus, when a vector **p** is presented, the weights of the winning neuron *and* its close neighbors move toward **p**. Consequently, after many presentations, neighboring neurons have learned vectors similar to each other.

4. Introduction to Principal Component Analysis

Principal component analysis (PCA), which is also known as the Karhunen-Loève transformation, is perhaps the oldest and best-known technique in multivariate analysis. It was first introduced by Pearson, who used it in a biological context. It was then developed by Hotelling in work done on psychometry. It appeared once again quite independently in the setting of probability theory, as considered by Karhunen and was subsequently generalized by Loève.

Principal component analysis can be considered one of the simplest forms of unsupervised learning when the manifold to fit is a linear subspace. Principal components are also used for initialization in more sophisticated unsupervised learning methods. The analysis is motivated by the following two problems.

1. Given a random vector $\mathbf{X} \in \mathbb{R}^d$ find the d'-dimensional linear subspace that captures most of the variance of **X**. This is the problem of feature

extraction where the objective is to reduce the dimension of the data while retaining most of its information content.

2. Given a random vector $\mathbf{X} \in \mathbb{R}^d$ find the d'-dimensional linear subspace that minimizes the expected distance of $\mathbf{X}$ from the subspace. This problem arises in the area of data compression where the task is to represent the data with only a few parameters while keeping low the distortion generated by the projection.

Consider a d-dimensional random vector $\mathbf{X} = (X_1, \ldots\ldots X_d)$ with finite second moments and zero mean. Let $\mathbf{u} \in \mathbb{R}^d$ be an arbitrary unit vector, and $\mathbf{s}(t) = t\mathbf{u}$ the corresponding straight line. Let $Y = t\mathbf{s}(\mathbf{X}) = \mathbf{X}^T\mathbf{u}$ be the projection index of $\mathbf{X}$ to $\mathbf{s}$. From $E[\mathbf{X}] = 0$ it follows that $E[Y] = 0$, and so the variance of Y can be written as

$$\sigma^2 Y = E[(\mathbf{X}^T\mathbf{u})^2] = E[(\mathbf{u}^T\mathbf{X})(\mathbf{X}^T\mathbf{u})]$$
$$= \mathbf{u}^T E[\mathbf{X}\mathbf{X}^T]\mathbf{u} = \mathbf{u}^T\mathbf{R}\mathbf{u}$$
$$= \Psi(\mathbf{u})$$

where the d x d matrix $\mathbf{R} = E[(\mathbf{X}\text{-}E[\mathbf{X}])(\mathbf{X}\text{-}E[\mathbf{X}])^T = E[\mathbf{X}\mathbf{X}^T]$ is the covariance matrix of $\mathbf{X}$. Since $\mathbf{R}$ is symmetric, $\mathbf{R} = \mathbf{R}^T$.

The principal directions along which the projection variance is stationary are the eigenvectors of the covariance matrix $\mathbf{R}$, and the stationary values themselves are the eigenvalues of $\mathbf{R}$. It also implies that the maximum value of the projection variance is the largest eigenvalue of $\mathbf{R}$, and the principal direction along which the projection variance is maximal is the eigenvector associated with the largest eigenvalue.

The first principal component line maximizes the variance of the projection of X to a line among all straight lines. The first principal component line minimizes the distance function among all straight lines, which represented as arg $\min_{\|u\|=1} \Psi(\mathbf{u}) = \mathbf{u}_1$.

5. Data Modelling for Web Usage Mining

Web log files are extracted by three sources of servers; they are from client server side, server side and from proxy server. The first part includes the domain dependent processes of transforming the Web data into suitable transaction form. That is by data preprocessing methods. Before executing the data mining algorithms we need to preprocess the web log data. The process includes data cleaning, user identification, session identification and path identification. Data Cleaning is web site based and involves tasks such as, removing extraneous references to embedded objects that may not be important for the purpose of analysis, including references to style files, image, or sound files. The cleaning process also may involve the removal of at least some of the data fields (e.g. number of bytes transferred or version of HTTP protocol used, etc.) which may not provide useful information in analysis or data mining tasks. Data cleaning also entails the removal of references due to crawler navigations.

Web log data has both qualitative and quantitative data. For our analysis, We use Multiple Correspondence Analysis on a set of navigations related to the web link data. To obtain some groups of navigations and their interpretations for this analysis, we need to structure together the set of Web pages in semantic topics. The algorithm that we used is again Diday's moving clouds method belongs to the family of the kmeans algorithms. The objective is to develop a strategy which analyses the relations between the structure of the Web site and the log file. To reach this, we apply two different hybrid clustering methods. The Analysis of web log file links navigations are continuous. This hybrid clustering algorithm uses Principal component analysis (PCA) for visualization of the correlations between variables defined in the log file and Dynamic Clustering Method on the principal factors given by PCA with SOM and neural gas Algorithm.

5.1. Proposed SONG-PCA Clustering Algorithm

INPUT

There are n-dimensional input data say , $\{x(t)\}$ t =1, 2, …..l , the mean vector **e** and the covariance matrix of $x(t)$ are defined by

$$\mathbf{e} = \frac{1}{l}\sum_{t=1}^{l} x(t),$$

$$\mathbf{R} = \frac{1}{l-1}\sum_{t=1}^{l}[(x(t)-\boldsymbol{e}),(x(t)-\boldsymbol{e})^T].$$

Neural gas algorithm with Euclidean distance is as follows.

Step 1: Initialization:

Define n-neurons, namely the number of clusters.
Weigh vectors over the input space, W = [w_1, w_2,.....w_c].
Initial learning Rate η_0 and the final learning rate η_{end}.
Number of training set N and the maximum training epoch EP with $EP_0 = 0$ and $EP_{max} = M$.
Maximun number of iterations $t_{max} = MN$

Step 2:Processing:

Input vector x(t) at time t in m^{th} training epoch. Then total iterations are
t_{iter} = EP * N + t .

a) Calculate the distance (e.g., Euclidean distance)

$d_i = \|x(t) - w_i\|$, I =1,2,…… clusters

b) Calculate the neighborhood ranking r_i (initial r_i = 0 and i=1,2,….c) and i = 1,2….,clusters and j = i,…..clusters.

$$r_i = r_i + \begin{cases} 1, & d_i \geq d_j \\ 0, & otherwise \end{cases}$$

c) Calculate the error

$$\Delta E_i = E_{rj}\, F_i \qquad \text{Where}$$

$$F_i = \frac{1}{\|w_i - \eta\|^2} \sum_{N_\eta} \|w_l - \eta\|^2$$

d) Update the weight vectors w_i as

$$w_{i+1} = w_i + \eta(t) h_\lambda(r_i)(x(t) - w_i)$$

$$h_\lambda(r_i) = e^{\frac{r_i}{\lambda(t)}}, \qquad \text{where}$$

$$\eta(t) = \eta_0 \left(\frac{\eta_{end}}{\eta_0}\right)^{\frac{t}{t_{max}}} \qquad \text{and}$$

$$\lambda(t) = \lambda_0 \left(\frac{\lambda_{end}}{\lambda_0}\right)^{\frac{t}{t_{max}}}$$

Increase t = t+1.

Repeat Step 2 until $t = t_{max.}$

Step 3: Representation:

In the last training epoch, label the input $x(t)$ as one of the stabilized clusters according to the following criteria,
C_j = arg $\min_j\{r_j\}$, $j = 1,2, \ldots . c$

Note that the winning neuron C_j with minimum neighborhood ranking r is the Best-Matching-Unit with Euclidean sense [13].

The above algorithm describes the clustering based on self organising neural gas combined with principal component analysis. PCA has been widely used in data compression and feature selection. By sorting the eigenvalues λ_i of R in descending order, we obtain m eigenvectors, also called principal directions, corresponding to those m largest eigenvalues (usually $m<n$) for feature extraction.

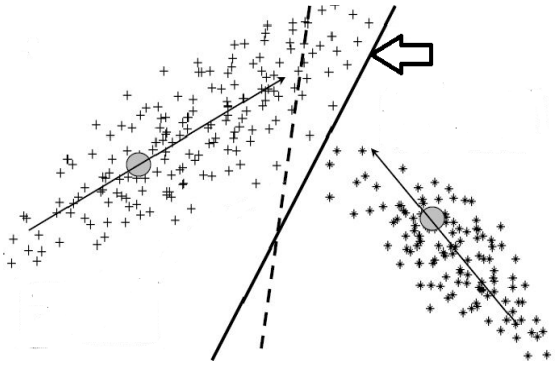

Fig 1. PCA with SONG Clustering

The above Fig.1 describes that there are two Gaussian based clusters and centroid of the cluster is denoted by a small solid circle. In a Euclidean sense, the separation line is vertical to the line which joints the centroids of the two clusters and passes through its midpoint. Obviously, there exist data samples in top cluster are misclassified by the Euclidean-metric-based separation line (the bold dashed line). But the separation line in a local principal subspace decomposition sense (the bold solid line denoted by an arrow) can match this Gaussian-based distribution data clusters well and can separate them distinctly (see Fig. 1). Where λ_i represents the corresponding eigen values (in a descending order) of covariance matrix R, and α is the proportion factor usually more than 90%.

6. Conclusion and Future Work.

We introduced a Self-Organizing Neural gas approach to the study of mining Web log data. Starting from the raw Web log data that is available in any Web server, we preprocessed it into distinct user transactions. We used the classical *k*-means algorithm to classify the URLs into clusters based on users' browsing history. The experimental results based on the data from the Web log of the server demonstrate that our approach is very useful is a specified domain. The results of the clusters generated form the SOM network shows that our approach can effectively discover usage patterns. Our results can also be used to predict the user's browsing behavior based on the past experience.

In this paper we proposed new clustering SONG-PCA Algorithm. The Self Organizing neural gas network is used in the input space. Kohonen's SOM is held in the output space. Combine with principal component analysis. It has some major features, which includes Euclidean distance measure, Error Calculation, Weight Adjustment, Neighborhood selection and Time based recognition. This algorithm approximates the data distribution.

Also we do have the idea of future implementation based on this proposed algorithm as follows. Experimental results with comparison analysis of all clustering algorithms using web log data. Adapting the SONG-PCA algorithm to large dimensionality input spaces like image compression etc., Insertion of new nodes to reduce errors to optimize the design.

7. References

1. Aguilar.J.S, Ruiz.R, Riquelme J.C and Gir´aldez.R, (2001) SNN:A Supervised Clustering Algorithm, in: 14th *Int. Conf. on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems (IEA/AIE 2001): Lecture Notes in Artificial Intelligence, Springer-Verlag*, Budapest, Hungary, June 4–7,2001, pp. 207–216.
2. Cooley.R. (2000) Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data. PhD thesis, University of Minnesota, May 2000.
3. Graham .J and Starzyk J, (2008) A Hybrid self Organizing Neural Gas Network , *IEEE World Conference on Computational Intelligence (WCCI'08) June 1-6*
4. Jirayusakul .A and Auwatanamongkol .S (2000) A supervised growing neural gas algorithm for cluster analysis.
5. Kohonen.T,(1995). Self-Organizing Maps, Berlin, Germany: *Springer*,
6. Martinet.Mz, Berkovich.S and Schulten.K, (1993). Neural-gas network for vector quantization and its application to time series prediction, *IEEE Trans. Neural Networks* 4(1993)558-569.
7. Pedrycz.W and Vukovich.G, (2004) Fuzzy clustering with supervision, Pattern Recognition **37**(7), 1339–1349.
8. Qu.Y and Xu.S, (2004) Supervised cluster analysis for microarray data based on multivariate Gaussian mixture, Bioinformatics **20**(12), 1905–1913.
9. Slonim.N and Tishby.N, (1999) Agglomerative information bottleneck, in: *Proceedings of the 13th Neural Information Processing Systems*, (NIPS).
10. Sonali Muddalwar and Shashank Kawar, (2012) Applying Artificial neural network in Web Usage Mining, International Journal of Computer Science and Management Research Vol 1 Issue 4 November 2012.
11. Yu.F, Sandhu..K, and Shih .M. (2000) A generalization-based approach to clustering of web usage sessions. In Proc. of the 1999 KDD Workshop on Web Mining, San Diego, CA. *Springer-Verlag*, volume 1836 of LNAI, pages 21-38. Springer, 2000.
12. Zeidat.N, (2005) Supervised Clustering: Algorithms and Application, Ph.D Dissertation, University of Houston.
13. Xiufen Fang et al., A(2010) Principal Components Analysis Neural Gas Algorithm for Anomalies Clustering. *WSEAS TRANSACTIONS on SYSTEMS*. Vol 9 Issue 1 Jan 2010.