A ROBUST NETWORK APPLICATION ALGORITHM BASED ON THE FRAMEWORK OF NEAREST PROTOTYPE

A. Abhishek Reddy[1], B. Indira Priyadarshini [2], CH.Swathi[3]
[1, 2] Assistant Professor, ECE Dept., Matrusri Engineering college, Hyderabad
[3] Assistant Professor, ECE Dept., NRI Institute of Technology, Vijayawada
Email:Abhishek.a.reddy@gmail.com[1], priyadarshini414@gmail.com[2],
swathichalumuri6@gmail.com[3]

Abstract

Network traffic is the key predicament and has gained much importance in the field of networking. This paper primarily discusses the two clustering methods used in the analysis of the network traffic. A comprehensive data processing is performed and application profiles are produced and consecutively clusters from the network traffic data, stating the similarities between different applications, which in turn are used to manage the network resources. This paper predominantly contemplates with enhanced Nearest Prototype (NP) classification algorithm.

Index terms: clustering, nearest prototype, network, traffic, Resources.

INTRODUCTION

Efficient network management of network resources has become a complicated task with rapid increase in the number of network communication applications, protocols and end-users and traditional management methods are in vain in obtaining a comprehensive view of the state of the network and simultaneously discover important details from the traffic. Data mining caters to these needs. Buoyant results of using rough sets [1] or LVQ [2] methods for similar purposes are there to discover the key characteristics and regularities within the network traffic. The SOM (Self Organizing Map) has been premeditated in finding fraud user profiles for cellular phone [3]. Neural networks recognize a limited set of applications. The self-organizing map (SOM) and the fuzzy C-means (FCM) clustering algorithm were chosen to be used in the analysis because of their robust and illustrative nature.

The nearest neighbor (NN) rule is one of the most successfully used techniques for resolving classification and pattern recognition tasks. Despite its high classification accuracy, this rule suffers from several shortcomings in time response, noise sensitivity and high storage requirements. These weaknesses have been tackled from many different approaches, among them, a good and well-known solution that we can find in the literature consists of reducing the data used for the classification rule (training data). Prototype reduction techniques can be divided into two different approaches, known as prototype selection and prototype generation or abstraction. The former process consists of choosing a subset of the original training data, whereas prototype generation builds new artificial prototypes to increase the accuracy of the NN classification. This technical report provides a survey of the prototype generation methods proposed in the literature.

HYPOTHETICAL BACKDROP

Fuzzy C-means (FCM) algorithm, also known as Fuzzy ISODATA, was introduced by Bezdek [6] as an extension to Dunn's algorithm [7]. The FCM-based algorithms are the most widely

used fuzzy clustering algorithms in practice. Fuzzy C-means (FCM) algorithm, also known as Fuzzy ISODATA, was introduced by Bezdek [6] as an extension to Dunn's algorithm [7].

Let X= {x1, x2,…, xN}, where xi∈ ℜn present a given set of feature data. The objective of FCM-algorithm is to minimize the Fuzzy C-Means cost function formulated as

$$J(U,V) = \sum_{j=1}^{C}\sum_{i=1}^{N}(\mu_{ij})^{m}\|x_i - v_j\|^2$$

$V = \{v_1, v_2, \ldots, v_C\}$ are the cluster centers.

$$U = (\mu_{ij})_{N\times C}$$

is a fuzzy partition matrix, in which each member μij indicates the degree of membership between the data vector xi and the cluster j. The values of matrix U should satisfy the following conditions

$$\mu_{ij} \in [0,1], \forall i = 1,\ldots,N, \forall j = 1,\ldots,C$$

$$\sum_{j=1}^{C}\mu_{ij} = 1, \forall i = 1,\ldots,N$$

The exponent m ∈ [1, ∞] is the weighting exponent, which determines the fuzziness of the clusters. The most commonly used distance norm is the Euclidean distance $d_{ij} = \|x_i - v_j\|$, although Babushka suggests that other distance norms could produce better results [8]. Minimization of the cost function J(U,V) is a nonlinear optimization problem, which can be minimized with the following iterative algorithm:

Step 1: Initialize the membership matrix U with random values so that the conditions (2) and (3) are satisfied. Choose appropriate exponent m and the termination criteria.

Step 2: Calculate the cluster centers V according to the equation:

$$v_j = \frac{\sum_{i=1}^{N}(\mu_{ij})^{m}x_i}{\sum_{i=1}^{N}(\mu_{ij})^{m}}, \forall j = 1,\ldots,C$$

Step 3: Calculate the new distance norms:

$$d_{ij} = \|x_i - v_j\|, \forall i = 1,\ldots N, \forall j = 1,\ldots,C$$

Step 4: Update the fuzzy partition matrix U:
If dij>0 (indicating that xi≠ vj)

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}}$$

Else
μij = 1

A suitable termination criterion could be to evaluate the cost function (Eq. 1) and to see whether it is below a certain tolerance value or if its improvement compared to the previous iteration is below a certain threshold [9]. Also the maximum number of iteration cycles can be used as a termination criterion.

SELF-ORGANIZING MAP

The self-organizing map (SOM) [10], first introduced by Kohonen, is a powerful clustering and data presentation method. A SOM consists of a grid shaped set of nodes. Each node j contains a prototype vector mj ∈ ℜn, where n is also the dimension of the input data.

SOM is trained iteratively. For each sample xi of the training data, the best matching unit (BMU) in the SOM is Located

$$c = \arg\min_{j}\{\|x_i - v_j\|\}$$

Index c indicates the corresponding prototype (BMU) vector mc. The Euclidian norm is usually chosen for the distance measure. The BMU mc and its neighboring prototype vectors in the SOM grid are then updated towards the sample vector in the input space

$$m_j(t+1) = m_j(t) + h_{cj}(t)[x_i - m_j(t)]$$

According to the neighborhood function the above equation states each prototype is turned towards the sample vector according to the neighborhood function. The figure below shows the comparison analysis of the different modeled algorithms.

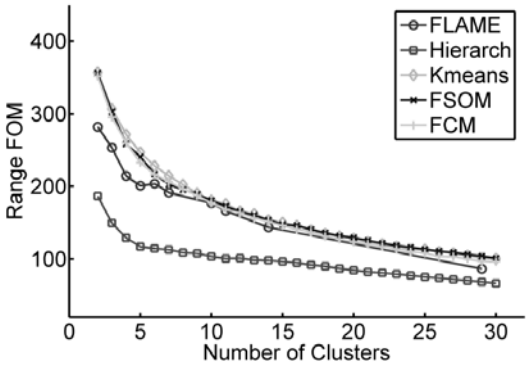

Figure 1. Comparative study of various optimization techniques.

INVESTIGATIONS & RESULTS

The statistical information of the traffic is provided by the Net Flow network management data. A flow is defined as unidirectional sequence of packets between given source and destination endpoints [11]. Different endpoints are separated with source and destination IP addresses, port numbers, protocol and ToS numbers. A session is a collection of flows with same endpoints. The measured NetFlow network data consisted of over 274000 samples of different computer application sessions, gathered from an edge router of a LAN network. The LAN was used for test purposes and contained several types of users and services. The arrangement allowed us only to measure traffic one way, from the LAN network to Internet. All the data preprocessing and analysis was done with Matlab software. The SOM toolbox used in the analysis is available in [12]. The original data was first preprocessed and the essential attributes were selected for the cluster analysis. Some attributes had to be further formed from the original Net- Flow features. Selected attributes for each session were Number of packets Bytes/Packet ratio, Duration of session in seconds, Feature indicating whether the dominating application was used as a source or destination port, indicating the direction of the first flow, Time from previous similar session between the same endpoints (max. 3600 sec.), Application (TCP/UDP port) number. The application number was not used in the actual clustering process, since it is a categorical attribute and as such can not be used with continuous clustering methods. But it can be used for labeling clusters with both of the used methods. For the final analysis, 5 weekdays of traffic data was chosen, total sum of 27801 sessions. In future research projects the data amount used for analysis will be much higher. The data was divided into training and validation sets, containing 75% and 25% of the total amount, respectively. Most of the applications were only used few times in the 5 day period. Only the 20 most frequently used applications were therefore selected. Also the data had to be normalized, because of the different ranges in values between attributes. SOM and FCM algorithms were run several times to make sure that the randomized initializations would not affect the results. According to the Xie-Benin [13] index the number of the clusters for FCM was chosen to be 25. The size of the SOM map was made large, so that even the small application clusters would be clearly visible. The U-matirx with application numbers used as labels and comparison of SOM and FCM clustering results SOM prototype vectors are labeled with FCM cluster numbers considering the data [1] are shown in the figures below.

RESULTS

The SOM map was 40 by 22 nodes in size. Prototype vectors were labeled using the validation data set. Figures above illustrate the U-matrix of the labeled SOM map. In U-matrix the darker tone implies that the neighboring prototype vectors are more similar with each other than those in the whiter areas of the map. Map clearly has some well separated clusters, for example the top right corner area containing labels 8888 and the lower left corner with a dense cluster of labels 0. The DNS application has again numerous labels scattered around the U-matrix. U-matrix also reveals how similar the DNS, HTTP and ICMP applications actually are.

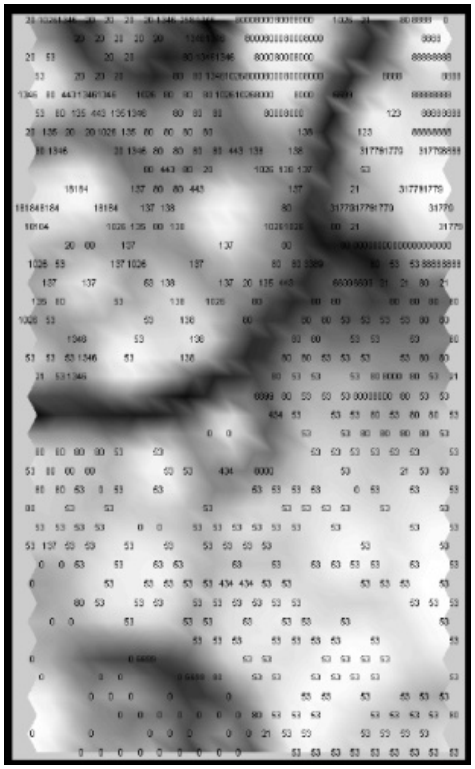

Figure 2. Result of SOM mapping . (a) U – Matrix -1

Figure 2. Result of SOM mapping. (b) U – Matrix -2

CONCLUSION

One of the important research topics on minimum distance classification is prototype optimization. Using a few, but well optimized, prototypes, it is possible to achieve an even better classification performance in both speed and accuracy than using all the training samples. In this paper, a novel error function containing class classification and class separation components has been proposed. Not only prototype but also feature weight update rules have been accordingly derived. The physical meanings of these update rules have also been discussed in detail, which reveals that the proposed method consists of several novel learning properties, such as robustness to outlier samples and effectiveness in learning feature weights. The analyzed data bestows valuable information of the network state compared to the actual human generated traffic. The closer the applications are located on the map grid, the more similar they are in characteristics. Same as FCM, SOM map concentrates more on the areas where the data density is higher. Applications with less sample vectors are easily left with just a few prototype vectors in the map. However the rare applications were usually present in the map which makes it possible to see their relation with the other applications. Comparison between the two methods was done by applying the modified NP classification procedure combined with the U-matrix. The produced U-matrix showed supported the correctness of both of the methods. This procedure proved out to be a prominent way to compare and validate any clustering results to those obtained with SOM map. The obtained clusters will eventually be used for application profiling. The application profiles combined with the knowledge of the user behavior and the topological information of the network are essential in producing the dynamical model for the computer network. Application resource requirements will help the network administrators and the service providers to anticipate, what kind of changes new users and applications will cause in use of network resources.

REFERENCES

[1] V. Laamanen, M. Laurikkala and H. Koivisto, "Network Traffic Classification Using Rough Sets", Recent Advances in Simulation, Computational Methods and Soft Computing, WSEAS Press, 2002.

[2] M. Ilvesmäki and M. Luoma, "On the capabilities of application level traffic measurements to differentiate and classify Internet traffic", Internet Perfomance and Control of Network Systems II, Proceedings of SPIE, Vol. 4523, 2001.

[3] J. Hollmén, User profiling and classification for fraud detection in mobile communication networks, Doctorate thesis, Helsinki University of Technology, Finland, 2000.

[4] K.M.C. Tan and B.S. Collie, "Detection and classification of TCP/IP Network services", Proceedings of the 13th Annual Computer Security Applications Conference, USA, 1997.

[5] K.C. Claffy, "Measuring the Internet", IEEE Internet Computing, Vol. 4, no. 1, pp. 73-75, 2000.

[6] J. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, USA, 1981.

[7] J.C. Dunn, "A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact, Well Separated Clusters", Journal of Cybernetics, Vol. 3, No. 3, pp. 32-57, 1974.

[8] R. Babuska, Fuzzy Modelling for Control, Kluwer Academic Publishers, The Netherlands, 1998.
[9] J.-S. Jang, C.-T. Sun and E. Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall, USA, 1997.
[10] T. Kohonen (Editor), Self-Organizing Maps, Springer-Verlag, Germany, 2001.
[11] Cisco Systems, Inside Cisco IOS Software Architecture, Cisco Press, USA, 2000.
[12]SOMtoolbox,
http://www.cis.hut.fi/projects/somtoolbox/,
Referenced 13th of June 2002.
[13] X.L. Xie and G. Beni, "A validity measure for fuzzy clustering", Transactions on Pattern Analysis and Machine Intelligence. Vol. 13, no. 8, pp. 841-847, 1991.