Characterization of Network Traffic Data

A Data Preprocessing and Data Mining Application

Esra Kahya-Özyirmidokuz, Ali Gezer, Cebrail Çiflikli

Kayseri Vocational College, Erciyes University, Kayseri, Turkey

e-mails: {esrakahya, aligezer , cebrailc} @erciyes.edu.tr

Abstract— **Large amount of traffic data are transmitted during day-to-day operation of wide area networks. Due to the increment of diversity in network applications, its traffic features have substantially changed. Data complexity and its diversity have been rapidly expanding with the changing nature of network applications. In addition, bandwith and speed of network have increased rapidly as compared to the past. Therefore, it is a necessity to characterize the changing network traffic data to understand network behavior. The aim of this research is to understand the data nature and to find useful and interesting knowledge from the network traffic traces which contains IP protocol packets. We analyze the traffic trace of 21 April 2012 on a 150 Mbps transpacific link between US and Japan from the MAWI Working Group traffic archive. This data contain lots of useful and important information which is hidden and not directly accessible. In this research, firstly, anomaly detection analysis and Kohonen Networks are applied to reduce the data matrix. Then, we generate a CART decision tree model to mine traffic data. The decision tree method is successfully applied in network traffic analysis. The results show that the proposed method has substantially good performance.**

Keywords-Network traffic data analysis; Kohonen networks; Data mining; CART; Preporcessing process.

I. INTRODUCTION

Network applications have substantially differed in recent years. Previously, server-client application traffic constitutes most of the Internet traffic. Nowadays, peer-to-peer protocols and applications take a large amount of the total bandwith on the Internet [2]. Also, due to the continuous growth of network speed and bandwith, large amount of traffic are transmitted during day-to-day operation of wide area networks. Therefore, data complexity and its diversity have been rapidly expanding as the increased amount of traffic and changing nature of applications. The characterization of the network traffic data becomes a necessity to understand network behavior. Our objective is using data mining techniques, specifically decision trees, to understand the data characterization and to find useful and interesting knowledge from the network traffic which contain IP protocol packets. We analyzed the traffic trace of 21 April 2012 taken from MAWI Working Group traffic archive. The trace was captured on a 150 Mbps transpacific link between US and Japan. Capturing starts at 2 p.m. and finishes at 2:15 p.m. Due to the large amount of captured packets, we use only 1226014 packets in our analysis. This data contain lots of useful and important information which is hidden and not directly accessible.

After building the database, exploratory data analysis is performed. An approach is presented for pre-processing the data for improving the quality of data and removal of noisy, erroneous and incomplete data. Believability of the data is controlled. Incomplete and inconsistent data analysis are applied. Moreover, anomaly detection analysis is used to reduce the records of data matrix in the data preprocessing process. Although there are many different techniques which can be used in this study, e.g., PCA [20][25], factor analysis [10] and attribute relevance analaysis [3][4], we used Kohonen Networks (KN) in clustering due to the strength of Kohonen maps that lies in their ability to model non-linear relationships between data. In addition, factor analysis, in its forms (PCA, CA, MCA) is the ideal method for providing an overview of the data and continuous spatial visualization of the individuals, and even, in some cases, detecting the natural number of clusters [23]. Kohonen Maps are useful tools for the data mining (DM) models with large data sets. High-dimensional data is projected to a lower dimension representation scheme that can be easily understood. Besides, Kohonen Maps can be used to process qualitative variables as well as quantitative ones [23]. In addition, it can be preferred because of the amount of data matrix. In this research, KN is applied to reduce the database.

There are numerous advantages of hierarchical classifiers based on DTs [23]. We have generated a CART (Classification and Regression Trees) DT (Decision Tree) model in SPSS Clementine 10.1. Colasoft Capsa 6.0 packet sniffer application is used to filter IPV4 packets.

The rest of the paper is organized as follows. The next section presents the literature. In Section III, we give a formal definition of KNs. We describe CART DT in Section IV. Section V presents the application. We draw our conclusion in Section VI.

II. LITERATURE REVIEW

Clustering algorithms have been, and continue to be, widely used for network traffic data classification [11][21][26]. Eshghi et al. [8] compared tree cluster techniques: traditional clustering methods, Kohonen Maps and latent class models. Each methodology could lead to potentially different interpretations of the underlying structure of the data.

There have been many applications of KNs to DM. Larose [6] uses the cluster memberships in data mining. A CART DT model was run, to classify customers, as either churners or nonchurners. Malone et al. [12] demonstrated a trained SOM (Self-Organising Map) which could provide initial information for extracting rules that described cluster

 ISBN: 978-1-61208-242-4

boundaries. They used iris, monks and lungCancer data. Gomez-Carracedo et al. [17] applied Kohonen SOMs to perform pattern recognition in four datasets of roadside soils samples obtained in four sampling seasons along a year. They used CART as an objective variable selection step before the SOM grouping. Siriporn and Benjawan [18] represented an unsupervised clustering algorithm namely sIB, RandomFlatClustering, FarthestFirst, and FilteredClusterer that previously not been used for network traffic classification. Exported network traffic data can be used for a variety of purposes, including DM modeling. Tan et al. [22] conducted network packets analysis on application layer, analyzed network resource location and data size accurately, and studied the traffic characters of network redundancy. Palomo et al. [7] reported that the SOM which was successful for the analysis of highly dimensional input data in DM applications as well as for data visualisation in a more intuitive and understandable manner, had some problems related to its static topology and its inability to represent hierarchical relationships in the input data. To overcome these limitations, they generated a hierarchical architecture GHSOM that is automatically determined according to the input data and reflects the inherent hierarchical relationships among them. Apiletti et al. [5] designed the NETMINE framework which allowed the characterization of traffic data by means of DM techniques.Zaki and Sobh [14] presented an approach to observe network characteristics based on DM framework. They designed a database system and implemented it for monitoring the network traffic.

III. KOHONEN NETWORKS

KNs represent a type of SOM, which itself represents a special class of neural networks. The goal of awl-organizing maps is to convert a complex high-dimensional input signal into a simpler low-dimensional discrete map. Thus, SOMs are nicely appropriate for cluster analysis, where underlying hidden patterns among records and fields are sought. A typical SOM architecture is shown in Figure 1 [6].

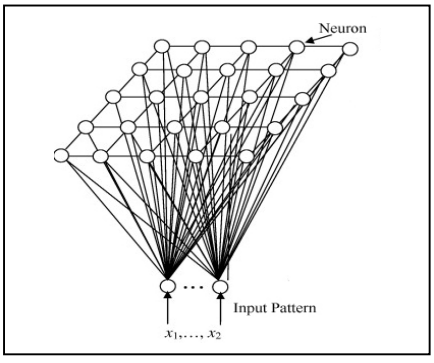

Figure 1. Topology of a simple SOM.

KNs can be considered a non-hierarchical method of cluster analysis. As non-hierarchical methods of clustering, they assign an input vector to the nearest cluster, on the basis of a predetermined distance function but they try to preserve a degree of dependence among the clusters by introducing a distance between them. Consequently, each output neuron has its own neighborhood, expressed in terms of a distance matrix. The output neurons are characterized by a distance function between them, described using the configuration of the nodes in a unidimensional or bidimensional space [19].

KNs perform unsupervised learning; an *Out* field is not specified and the model is, therefore, not given an existing field in the data to predict. KNs attempt to find relationships and overall structure in the data. The output from a Kohonen network is a set of (X, Y) coordinates, which can be used to visualize groupings of records and can be combined to create a cluster membership code. It is hoped that the cluster groups or segments are distinct from one another and contain records that are similar in some respect [24].

Suppose that we consider the set of *m* field values fot the *n*th record to be an input vector $x_n = x_{n1}, x_{n2}, \ldots x_{nm}$, and the current set of m weights for a particular output node *j* to be a weight vector $w_j = w_{1j,} w_{2j,\ldots} w_{mj}$. In Kohonen learning, the nodes in the neighborhood of the winning node adjust their weights using a linear combination of the input vector and the current weight vector:

$$w_{ij,new} = w_{ij,current} + \eta(x_{ni} - w_{ij,current}) \qquad (1)$$

where η, $0 < \eta < 1$, represents the learning rate, analogous to the neural networks case. Kohonen indicates the learning rate should be a decreasing function of training epochs (runs through the data set) and that a linearly or geometrically decreasing η is satisfactory for most purposes. The algorithm of KNs is shown in the accompanying box [6].

For each input vector *x*, do:

- Competition. For each output node j, calculate the value $D(w_j, x_n)$ of scoring function. For example, for Euclidean distance, $D(w_j, x_n) = \sqrt{\sum_i (w_{ij}, x_{ni})^2}$. Find the winning node *j* that minimizes $D(w_j, x_n)$ over all output nodes.
- Cooperation. Identify all output nodes j within the neighborhood of J defined by the neighborhood size R. For these nodes, do the following for all input record fields:
- Adaptation. Adjust weights: $w_{ij,new} = w_{ij,current} + \eta(x_{ni} - w_{ij,current})$
- Adjust the learning rate and neightborhood size, as needed.
- Stop when the termination criteria are met.

IV. DECISION TREES

The DT technique is one of the most intuitive and popular DM methods, especially as it provides explicit rules for classification and copes well with heterogeneous data missing, data and non-linear effects [23]. There are numerous advantages of hierarchical classifiers based on DTs. DTs provide an easy to understand overview for users without a DM background with high classification

accuracy. They also provide a tree model of the problem and various alternatives in an understandable format without explanation. The acquired knowledge are usually quite understandable and can be easily used to obtain a better understanding of the problem. In addition, DTs assist in making decisions with existing information. They have satisfactory performance even when the training data is highly uncertain.

A. CART Decision Tree

CART is a popular DT algorithm first published by L. Breiman, J. Friedman, R. Olshen, C. Stones in 1984 [13]. The CART algorithm grows binary trees and continues splitting as long as new splits can be found that increase purity. The CART algorithm identifies a set of such sub-trees as candidate models. These candidate sub-trees are applied to the validation set, and the tree with the lowest validation set misclassification rate (or average squared error for a numeric target) is selected as the final model [16].

The CART algorithm identifies candidate sub-trees through a process of repeated pruning. The goal is to prune first those branches providing the least additional predictive power per leaf. To identify these least useful branches, CART relies on a concept called the adjusted error rate. This is a measure that increases each node's misclassification rate or mean squared error on the training set, by imposing a complexity penalty based on the number of leaves in the tree. The adjusted error is used to identify weak branches (those whose error enough to overcome the penalty) and mark them for pruning. The formula for adjusted error rate is

$$AE(T) = E(T) + \alpha leaf\_count(T) \quad (2)$$

where α is an adjustment factor that is increased in gradual steps to create new subtrees. When α is 0, the adjusted error rate equals the error rate. The algorithm continues to find trees by adjusting α and pruning back one node at a time, creating a sequence of trees, α_1, α_2, and so on, each with fewer and fewer leaves. The process ends when the tree has been pruned all the way down to the root node. Each of the resulting subtrees (sometimes called alphas) is a candidate to be the final model. Notice that all the candidates contain the root node and the largest candidate is the entire tree [16].

The next step is to select, from the pool of candidate sub-trees, the one that works best on new data. That, of course, is the purpose of the validation set. Each of the candidate sub-trees is used to classify the records or estimate values in the validation set. The tree that performs this task with the lowest overall error is declared the winner. The winning sub-tree has been pruned sufficiently to remove the effects of overtraining, but not so much as to lose valuable information [16]. The winning sub-tree is selected on the basis of its overall error when applied to the validation set. But, while one expects that the selected sub-tree will continue to be the best model when applied to other data sets, the error rate that caused it to be selected may slightly overstate its effectiveness. There may be many sub-trees that all perform about as well as one selected. To a certain extent, the one of these that delivered the lowest error rate on the validation set may simply have "gotten lucky" with that particular collection of records. The selected sub-tree is applied to a third preclassified data set, the test set. The error obtained on the test set is used to predict expected performance of the model when applied to unclassified data [16].

CART uses the Gini index. The Gini impurity is,

$$I_E(m) = -\sum_{i=1}^{k(m)} \pi_i \log \pi_i \quad (3)$$

where π_i are the fitted probabilities of the levels present in node m, which are at most *k(m)* [19].

V. MODELING

This study uses the DM methodology CRISP-DM (Cross-Industry Standard Process for Data Mining) which is represented in Figure 2. According to CRISP-DM, a given DM project has a life cycle consisting of six phases as illustrated in Figure 2 [6].

A. Data Understanding

Preprocessing is an important step for successful DM to analyze the datasets before DM process starts. This section is focused namely on the second and third steps of Figure 2: data understanding and data preparation. The first step in the process is to transfer the data in a database, and make use of a statistical analysis tool to get the details of the network traffic data.

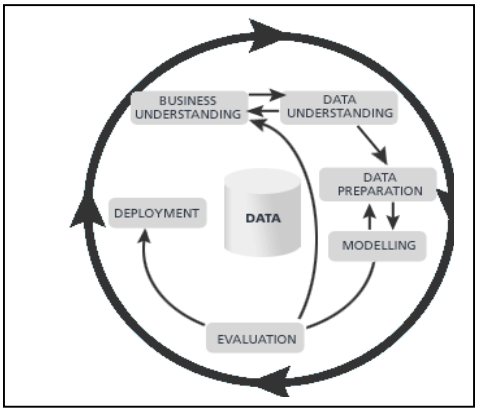

Figure 2. CRISP-DM is an iterative, adaptive process.

A large amount of data is used in this research. The trace data are obtained from MAWI traffic trace archieve. The trace data are captured on a 150 Mbps backbone line that connects US and Japan. Nearly 35 million packets and 31800000000 bytes are captured in 15 minutes time. In this analysis, we only use 1226014 packets captured on 21 April 2012. To filter IP protocol Colasoft Capsa 6.0 packet sniffer application is used.

A flow is identified as the combination of the following attributes which are presented and used in the data set are

{No, Date, Absolute time, Delta time, Relative time, Source, Destination, Protocol type, Size, IP identifier, Source physical, Destination physical, Source IP, Destination IP, Source port, Destination port, Comment, Summary}.

We eliminate date, comment, and summary attributes because they are useless and redundant in the model. The frequencies of protocol attribute variables are seen from Figure 3. In the preprocessing process, protocol attribute variables which are less than 0,01% are combined as 'others' variable. These updated variables are: MSSQL, POP3, RTCP, H.225, IMAP, BGP, QQ, SIP, POP3s, PPTP, MGCP, NFS, Telnet, H.323, PIM, LPD, SAP, WINS, IP, LDAP, PDM, NBDGM, NBSSN, SLP and Whols. SQL codes are used in this variable reduction. In the data matrix, there are 329 records whose protocol attribute variables are 'others'.

B. Anomaly Detection

DM uses huge amounts of data with many thousands or even millions of records, outliers and unusual data should be explored when preparing data for modeling. Anomalous data is a problem for models. In this research, anomaly detection analysis is used to find data values which show different behavior from the previous measured values. Anomaly detection is an important data mining task. It should be done in the data preprocessing process.

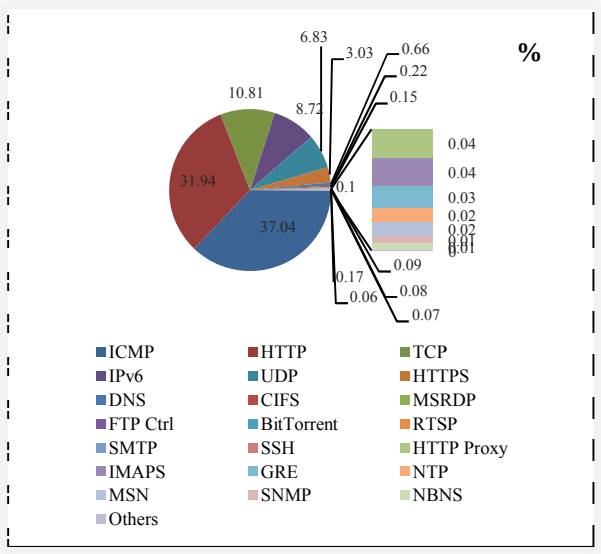

Figure 3. Protocol Attribute Variables

Anomaly detection models are used to identify outliers, or unusual cases in the data. Anomaly detection can examine large numbers of fields to identify clusters or peer groups into which similar records fall. Each record can then be compared to others in its peer group to identify possible anomalies. After anomaly detection analysis, an anomaly index is calculated for each record which is the ratio of the group deviation index to its average over the cluster that the case belongs to [3]. In this study, anomaly detection is used to locate the records that are the most unusual with respect to those fields. 12260 anomalous records are identified with eight clusters; that is the 1% of the cases that we requested. Average anomaly index level is 3.14041. Records which are greater than the average anomaly index level are selected as anomaly records. For example, record-1's protocol attribute value is "http". After anomaly detection, the record-1 which has "http" protocol value is determined in group 4. This record's anomaly index level is 1.079. We cannot eliminate the record because of its low index. 12260 of records are eliminated using the anomaly detection algorithm with the SPSS Clementine 10.1 DM Tool.

C. Data reduction via Kohonen Networks

Data reduction is an important stage for data preprocessing. The reduction technique is very important because as the data space is reduced, information can be lost. The reduced data must give a complete picture of the data to the analyst.

Cluster membership may be used to enrich the data set and improve model efficacy. Indeed, as data repositories continue to grow and the number of fields continues to increase, clustering has become a common method of dimension reduction [6].

KNs require us to specify the number of rows and the number of columns in the grid space characterizing the map. Large maps are usually the best choice, as long as each cluster has significant number of observations. The learning time increases significantly with the size of the map. The number of rows and the number of columns are usually established by conducting several trials until a satisfactory result is obtained [19].

The Kohonen SOM has several important properties that can be used within the DM/knowledge discovery and exploratory data analysis process. A key characteristic of the SOM is its topology preserving ability to map a multi-dimensional input into a two-dimensional form. This feature is used for classification and clustering of data. [12]. Standard clustering methods do not handle truly large data sets well, and fail to take into account multi-level data structures [1]. In this study, we use KNs to reduce the attributes of the clustering model. Therefore, only selected attributes are then kept to represent the document collection, the remaining ones are discarded.

Records are grouped by KN so that records within a group or cluster tend to be similar to each other, and records in different groups are dissimilar. The Kohonen parameters were set in SPSS Clementine 10.1 as follows. For the first 20 cycles, the neighbourhood size was set at R=2, and the learning rate was set to decay linearly starting at $\eta = 0.3$. Then, for the next 150 cycles, the neighborhood size was reset to R=1 while the learning rate was allowed to decay linearly from $\eta = 0.3$ to $\eta = 0$.

The neurons are organized into two layers, input layer and output layer. In the study, the input layer has 6 neurons and the output layer has 70 neurons.

 ISBN: 978-1-61208-242-4

The most attractive feature of the Kohonen SOMs is that, once trained, the map represents the projection of the data set belonging to an N-dimensional space into a bi-dimensional one [9]. The KN performs the projection of the H–dimensional space, containing the X vectors representing the load diagrams, into a bi-dimensional space. When a data stream passes through the generated model Kohonen node, two new fields are created, representing X- and Y-coordinates of the clusters. The clusters are identified in the Kohonen output window by their values on these coordinates [24]. Two coordinates $KX(10) and $KY(7) are representing the Kohonen net attributes.

A nine by six Kohonen clustering is performed. The unwanted attribute absolute time, destination and source attributes are blocked, leaving only 8 attributes. Delta time, relative time, IP identifier, destination physical, source physical, and size attributes and Kohonen-X and Kohonen-Y are selected as important attributes. The elimination of absolute time is expected because absolute time attribute has a direct relationship with delta time and relative time attributes.

D. Data mining modeling

There are many different types of classification methods. The choice of the best predicting technique depends on the data set being analyzed, and its complexity, the time it takes to generate, and the results of the analysis.The output attribute is discrete. In addition, DTs are powerful and popular tools for classification and prediction.

In this section, we present data mining modeling phase which is the fourth step of Figure 2. purpose of the decision tree is to provide a simple and understandable model of data. A CART DT model is run to classify network traffic data to build a DT for predicting network protocol type. The aim of the classification is to find the similar data items which belong to the same class.

Gini index is used. The depth of the tree is limited to 5 levels below the root node. The stopping criteria details are as follows: the minimum records in parent branches are 4%, and the minimum records in child branches are 2%. The minimum change in impurity is 0,0001. Maximum surrogates are 5.

In the CART model, the target is the protocol attribute. Delta relative size, IP identifier, source physical and destination physical are the input attributes. The model is given in Figure 4. We can show the tree model with if-then rules to express the process in English. 13 rules are generated. The following examples illustrate some of the rules:

- If [ip identifier=<46.5] and [size<=1.499] then [protocol is ipv6].
- If [ip identifier>46.5] and [size>64.5] and [size<=70.5] then [protocol is TCP].

E. Accuracy

The cross-validation method involves partitioning the examples randomly into *n* folds. (Ten is a fairly popular choice for *n*, but much depends on the number of examples available). We use one partition as a testing set and use the remaining partitions to form a training set. As before, we apply an algorithm to the training set and evaluate the resulting model on the testing set, calculating the percentage correctly. We repeat this process by using each of the partitions as the testing set and using the remaining partitions to form a training set. The overall accuracy is the accuracy averaged over the number of runs, which is equivalent to the number of partitions. *Stratified* cross-validation involves creating partitions so that the number of examples of each class is proportional to the number in the original set of examples [15].

```
IP Identifier <= 46,500 [ Mode: ICMP ]
    Size <= 74,500 [ Mode: ICMP ] => ICMP
    Size > 74,500 [ Mode: IPv6 ] => IPv6
IP Identifier > 46,500 [ Mode: HTTP ]
    Size <= 32,759 [ Mode: HTTP ]
        Size <= 1,508 [ Mode: HTTP ]
            Size <= 1,498 [ Mode: HTTP ]
                Size <= 1,394 [ Mode: UDP ] => UDP
                Size > 1,394 [ Mode: HTTP ] => HTTP
            Size > 1,498 [ Mode: IPv6 ] => IPv6
        Size > 1,508 [ Mode: HTTP ]
            Source Physical in [ "00:0E:39:E3:34:00" ] [ Mode: HTTP ]
                IP Identifier <= 24.484 [ Mode: HTTP ] =>
                HTTP
                IP Identifier > 24.484 [ Mode: HTTP ] =>
                HTTP
            Source Physical in [ "00:16:9C:7C:B0:00" ] [ Mode: HTTP ]
                => HTTP
    Size > 32,759 [ Mode: ICMP ]
        Source Physical in [ "00:0E:39:E3:34:00" ] [ Mode: HTTP ]
            Size <= 70,500 [ Mode: HTTP ] => HTTP
            Size > 70,500 [ Mode: UDP ]
                Size <= 188,500 [ Mode: UDP ] => UDP
                Size > 188,500 [ Mode: HTTP ] => HTTP
        Source Physical in [ "00:16:9C:7C:B0:00" ] [ Mode: ICMP ]
            Size <= 64,500 [ Mode: ICMP ]
                IP Identifier <= 30602,500 [ Mode: ICMP ] =>
                ICMP
                IP Identifier > 30602,500 [ Mode: ICMP ] =>
                ICMP
            Size > 64,500 [ Mode: TCP ]
                Size <= 70,500 [ Mode: TCP ] => TCP
                Size > 70,500 [ Mode: ICMP ] => ICMP
```

Figure 4. Data Mining Model

Tenfold cross-validation accuracy evaluation is used to train and test the data matrix. Records (917984) of data (1213754) are correct. The accuracy ratio of the model is 75.63% which is shown in Table III.

TABLE I. MODEL ACCURACY

	Number	Ratio
Correct	917984	75,63%
Incorrect	295770	24,37%

VI. CONCLUSION AND FUTURE WORK

DM is finding wide application in many fields. Network traffic characterization is one of them. Due to the increasing

diversity of network applications, their packet features substantially have changed. Also the growth in network speeds and bandwidths increases the amount of traffic on networks. Therefore, it is a necessity to characterize new traffic features to handle network problems and to increase performance.

In this study, we presented knowledge induction from network traffic data captured on a 150 Mbps trans-Pacific line between Japan and US. Preprocessing techniques were used to improve the quality of data. Noisy, erroneous, and incomplete data were removed from the data matrix. Anomaly detection analysis was used to reduce the data matrix. Moreover, this research included attribute reduction using KNs. A CART DT which uses for classification of the data set with five tree depths is generated. The accuracy ratio of the model is 75.63%.

In this research, network traffic data are mined and useful relationships, groupings, associations are discovered. The DT model lay out the problem clearly so that all options can be explored. This acquired knowledge will be used to predict the future behaviors of the line. In addition, the DT model helps network operators to understand the behavior of network users. This research is also important to assess future network capacity requirements and to plan future network developments.

For the future we plan the further evaluation and implementation of this framework.

REFERENCES

[1] A. Ciampi and Y. Lechevallier, "Clustering large, multi-level data sets: an approach based on Kohonen self-organizing maps", Principles of Data Mining and Knowledge Discovery 4th European Conference PKDD 2000 Proceedings Lecture Notes in Artificial Intelligence, vol. 1910, Springer-Verlag, Berlin, Germany, pp. 353-358, 2000.

[2] C. Çiflikli, A. Gezer, A.T. Özşahin, and Ö. Özkasap, "BitTorrent packet traffic features over IPv6 and IPv4", Simulation Practice and Theory, vol. 18, iss. 9, October 2010, pp. 1214-1224.

[3] C. Ciflikli and E. Kahya-Özyirmidokuz, "Enhancing product quality of a process", Industrial Management and Data Systems, vol. 112, iss.8, pp. 1181-1200, 2012.

[4] C. Ciflikli and E. Kahya-Özyirmidokuz, "Implementing a data mining solution for enhancing carpet manufacturing productivity", Knowledge-Based Systems, vol. 23, 2010, pp. 783–788.

[5] D. Apiletti, E. Baralis, T. Cerquitelli and V. D'Elia, "Characterizing network traffic by means of the NETMINE framework", Computer Networks, vol. 53, pp. 774–789, 2009.

[6] D. T. Larose, Discovering Knowledge in Data: An Introduction to Data Mining. Hoboken, NJ, USA: Wiley. 2005.

[7] E. J. Palomoa, J. North, D. Elizondo, R.M. Luquea and T. Watson, "Application of growing hierarchical SOM for visualisation of network forensics traffic data", Neural Networks, vol. 32, pp. 275-284, 2012.

[8] Eshghi, D. Haughton, P. Regrand, M. Skaletsky, and S. Woolford, "Identifying groups: A comparison of methodologies", Journal of Data Science, vol. 9, pp. 271-291, 2011.

[9] F. Rodrigues, J. Duarte, V. Figueriredo, Z. Vale, and M. Cordeiro, "A comparative analysis of clustering algorithms applied to load profiling", Machine Learning and Data Mining in Pattern Recognition, Lecture Notes in Computer Science, vol. 2734, pp. 73-85, 2003.

[10] H. F. Wang and C. Y. Kuo, "Factor analysis in data mining", Computers & Mathematics with Applications, vol. 48, iss: 10-11, pp. 1765-1778, 2004.

[11] J. Erman, M. Arlitt, and A. Mahanti, "Traffic classification using clustering algorithms, in: MineNet '06, ACM Press, New York, NY, USA, 2006, pp. 281–286.

[12] J. Malone, K. McGarry, S. Wermter, and C. Bowerman, "Data mining using rule extraction from Kohonen self-organising maps", Neural Comput& Applic, vo.15, pp. 9-17, 2005.

[13] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, Classification and Regression Trees, Chapman and Hall/CRC. 1984.

[14] M. Zakia and T. S. Sobhb, "NCDS: Data mining for discovering interesting network characteristics", Information and Software Technology, vol. 47, pp. 189–198, 2005.

[15] M. A. Maloof, Machine Learning and Data Mining for Computer Security, Methods and Applications. USA: Springer-Verlag. 2006.

[16] M. J. A. Berry and G.S. Linoff, Data Mining Techniques for Marketing, Sales, and Customer Relationship Management, Third Ed.. NY: Wiley. 2011.

[17] M. P. Gomez-Carracedo, J.M. Andrade, G.V.S.M. Carrera, J. Aires-de-Sousa, A.Carlosena, and D. Prada, "Combining Kohonen neural networks and variable selection by classification trees to cluster road soil samples", Chemometrics and Intelligent Laboratory Systems, vol. 102, pp. 20-34, 2010.

[18] O. Siriporn and S. Benjawan, "Anomaly detection and characterization to classify traffic anomalies, Case Study: TOT Public Company Limited Network", World Academy of Science, Engineering and Technology, vol. 48, pp. 407-415, 2008.

[19] P. Giudici, Applied data mining. England: Wiley, 2003.

[20] Q. Guo, W. Wu, D.L. Massart, C. Boucon and S. de Jong, "Feature selection in principal component analysis of analytical data", Chemometrics and Intelligent Laboratory Systems, vol. 61, pp. 123-132, 2002.

[21] Q. Wang, V. Megalooikonomu, A clustering algorithm for intrusion detection, Proceedings of SPIE 5812, pp. 31–38, 2005.

[22] S. Tan, M. Chen, G. Yang, and Y. Wang, "Research on Network Data Mining Techniques", Energy Procedia, Singapore, vol. 13, pp. 4853 – 4860, 2011 [ESEP 2011, 9-10 December 2011].

[23] S. Tuffer, Data Mining and Statistics for Decision Making. Wiley, 2011.

[24] SPSS Inc., Introduction to Clementine and Data Mining, Chicago, 2003, http://homepage.univie.ac.at/marcus.hudec/Lehre/WS%202006/Methoden%20DA/IntroClem.pdf [retrieved: 08, 2012].

[25] W. Melssen, R. Wehrens and L. Buydens, "Supervised Kohonen networks for classification problems", Chemometrics and Intelligent Laboratory Systems, vol. 83, pp. 99–113, 2006.

[26] Y. Guan, A. Ghorbani, and N. Belacel, "Y-Means: a clustering method for intrusion detection", Proceedings of Canadian Conference on Electrical and Computer Engineering, 2003, pp. 4–7.

 ISBN: 978-1-61208-242-4