International Journal of Soft Computing 10 (4): 274-278, 2015
ISSN: 1816-9503

Generalization of Minkowski Distance Metrics in Mixed Case Analysis for Web Intrusion Detection System

[1]K.G. Maheswari and [2]R. Anita
[1]Department of MCA, [2]Department of Electrical and Electronic Engineering,
Institute of Road and Transport Technology, Erode, TamilNadu, India

Abstract: The rapid growth of the web applications are resulted in severe security issues which gives out various classifications of attacks related with web usages. These attacks are generalized by different characteristics and methods to make the system vulnerable for the easy injection of threats. In this study, the mixed case analysis using distance metrics is designed to classify the various types of web attacks based on the severity of the vulnerability. The set of network and web related attributes are taken from the renowned datasets which is dynamically stored in the log server for the future reference. Hence, these datasets are extracted for the detection system by classifying the attack, instantaneously generates the classes of data clusters. These clusters are used for learning metric in mixed cases for analysing the web related attacks in the renowned datasets.

Key words: Threats, metrics, attacks, clusters, datasets

INTRODUCTION

The insufficiency of traditional security tools like antivirus in facing current attacks conducts to the development of Intrusion Detection Systems (IDS). IDSs search in the network traffic for malicious signature and then send an alarm to the user (Robertson *et al*., 2006; Roesch, 1999). Since, current IDSs are signature based they still unable to detect new forms of attacks even if these attacks are slightly derived from known ones. So, recent researches concentrate on developing new techniques, algorithms and IDSs that use intelligent methods like Neural Networks, Data Mining, Fuzzy Logic and Genetic algorithms.

In mathematics, there are multiple feature spaces that are useful in enormous problems. For example, Euclidian space or Euclidian distance is the most popular metric that we use in the real world or Mahalanobis distance that we use to demonstrate the distance between two places in the city. In classification and clustering problems, the majority of methods, assumes the Euclidian feature space to solve their problems and they evaluate their result in this space. In this study, the Combination of Manhattan and Euclidian distance are used which called Minkowski distance metrics is learning. The Minkowski distance is a metric on Euclidean space which can be considered as a generalization of both the Euclidean distance and the Manhattan distance.

Minkowski distance: The Minkowski distance of order p between two points:

$$P = (x_1, x_2, ..., x_n) \text{ and } Q = (y_1, y_2, ..., y_n) \in R^n \quad (1)$$

is defined as:

$$\left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p} \quad (2)$$

For $p \geq 1$, the Minkowski distance is a metric as a result of the Minkowski inequality. For $p<1$, it is not the distance between (0, 0) and (1, 1) is $2^{1/p}>2$ but the point (0, 1) is a distance 1 from both of these points. Hence, this violates the triangle inequality.

Minkowski distance is typically used with p being 1 or 2. The latter is the Euclidean distance while the former is sometimes known as the Manhattan distance. In the limiting case of p reaching infinity, we obtain the Chebyshev distance:

$$\lim_{x \to \infty} \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p} = \max_{i=1}^{n} |x_i - y_i| \quad (3)$$

Corresponding Author: K.G. Maheswari, Department of MCA, Institute of Road and Transport Technology, Erode, TamilNadu, India

Similarly, for p reaching negative infinity, we have:

$$\lim_{x \to -\infty} \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p} = \min_{i=1}^{n} |x_i - y_i| \quad (4)$$

The Minkowski distance can also be viewed as a multiple of the power mean of the component-wise differences between P and Q.

Literature review: The rapid developments in computer science and engineering have led to expediency and efficiency in capturing huge accumulations of data. The new challenge is to transform the enormous of data into useful knowledge for practical applications. An earlier general task in data mining is to extract outstanding features for the prediction. This function can be broken into two groups feature extraction or feature transformation and feature selection (Ingham *et al*., 2007). Feature extraction (for example, principal component analysis, singular-value decomposition, manifold learning and factor analysis) refers to the process of creating a new set of combined features (which are combinations of the original features).

On the other hand, feature selection is different from feature extraction because it does not produce new variables. Feature selection also known as variable selection, feature reduction, attribute selection or variable subset selection is a widely used dimensionality reduction technique which has been the focus of much research in machine learning and data mining and found applications in text classification, web mining and so on (Damashek, 1995). It allows for faster model building by reducing the number of features and also helps remove irrelevant, redundant and noisy features. This allows for building simpler and more comprehensible classification models with classification performance. Hence, selecting relevant attributes are a critical issue for competitive classifiers and for data reduction. In the meantime, feature weighting is a variant of feature selection. It involves assigning a real-valued weight to catch feature.

The weight associated with a feature measures its relevance or significance in the classification task (Kruegel and Vigna, 2003). Feature selection algorithms typically fall into two categories; feature ranking and subset selection. Feature ranking ranks the features by a metric and eliminates all features that do not achieve an adequate score (selecting only important features). Subset selection searches the set of possible features for the optimal subset. Feature Ranking Methods are based on statistics, information theory or on some function of classifier's outputs (Kruegel and Vigna, 2003). In statistics, the most popular form of feature selection is stepwise regression. It is a greedy algorithm that adds the best feature (or deletes the worst feature) at each round. The main control issue is deciding when to stop the algorithm. In machine learning, this is typically done by cross validation (Kruegel *et al*., 2005).

MATERIALS AND METHODS

The KDD data set are passed to each Intrusion Detection System and Voting algorithm is used to make decision based on each IDS System result. Each IDS has same structure that we will explain in next study. IDS have two phases: the train phase and test phase. In train phase, researchers sample from train data frequently and in each sampling, we change the selection probability of samples. Figure 1 illustrates the training phase. If $A = \{a_1, a_2, a_3, a_4, ..., a_n\}$ and $S = \{s_1, s_2, s_3, s_4, ..., s_k\}$ were the training set and samples set, respectively at the first step, each sample has equal probability 1/n, we sample K samples from train data and pass them to metric learner 1.

Then, we learn the learner and at the next step we decrement the probability of correctly classified samples and increase incorrectly classified samples. If the learner has C correctly classified samples and I error samples, the increment and decrement of probabilities are as follow: if C samples were classified correctly, its probability reduced by multiplying it by factor α. So, each IDS will predict the label of each sample as attack connection or normal connection with some certainty. In the classification part of Fig. 2, we use Minkowski algorithm to make final decision. As you see in Fig. 2, we assign a weight to each IDS result. We use proposed intelligent fuzzy cognitive map classification algorithm in Fig. 3 to generate the optimum class labels for the attacks in the renowned datasets.

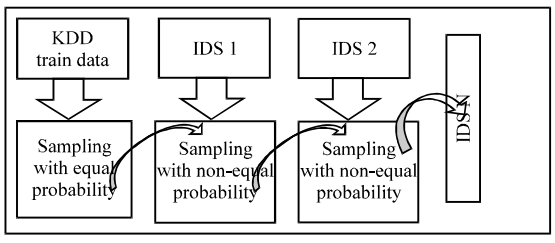

Fig. 1: Training phase of KDD dataset

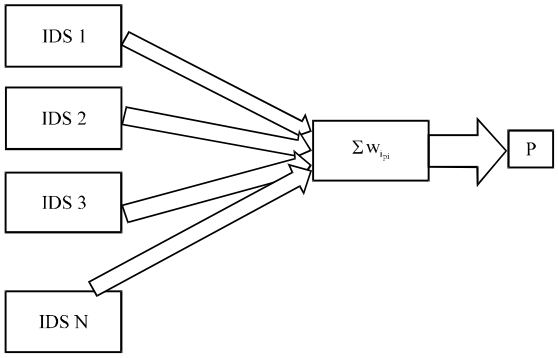

Fig. 2: Weight of each IDS

Input: featured set
Output: normal or abnormal
Step 1: confirm two initial centres and assign a new class c
Step 2: calculate minkowski distance between two classes

$$\sum_{i=1}^{n}|x(i)-y(i)|^{p}\Lambda\frac{1}{p}$$

Step 3: If (d(A, B)>d(A, C))
B is assigned as normal class
Else
C is assigned as attacker class
Step 4: Calculate min & max value

$$\lim_{p\to\infty}\left(\sum_{i=1}^{n}|x(i)-y(i)|^{p}\Lambda\frac{1}{p}\right)=\max_{i=1\to n}|x(i)-y(i)|$$

$$\lim_{p\to\infty}\left(\sum_{i=1}^{n}|x(i)-y(i)|^{p}\Lambda\frac{1}{p}\right)=\min_{i=1\to n}|x(i)-y(i)|$$

Step5: (A, B)<Threshold value, normal else attacker

Fig. 3: Intelligent Classification algorithm using Minkowski distance

Figure 3 explains the intrusion detection steps, till the nth model of detection (Cho and Cha, 2004). The KDD train data will calculate the equal sampling probability which enters to the 1st step of the IDS. The non-equal probability is given as an output till the nth model. The weight calculation of each IDS steps is explained by Fig. 2 in which the weight is calculated for individual IDS Model and then summation of those weight is given for the P which is defined as the sum of weight of IDS.

The above algorithm describes the classification technique of the attacks from the renowned datasets based on the minkowski distance metrics learning. The classification of attacks is based on the class labels. The class generation is defined by the rules of the minimum and maximum calculation of the distance values by the hybrid algorithms. The proposed system (Fig. 4) explains the Intrusion Detection System which starts from {1, 2, 3, ..., M). The IDS methods are analysed and classified using the proposed classification techniques.

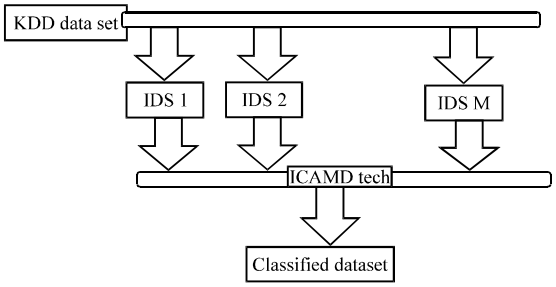

Fig. 4: Proposed system flow

RESULTS AND DISCUSSION

For the evaluation, we used 2 datasets, packets collected at a LAN gateway and the DARPA IDS evaluation dataset. The datasets are reconstructed into legitimate and illegitimate/attack accesses. We use HTTP instead of HTTPS encrypted traffic because there are few instances of attacks on HTTPS and an unencrypted attack can also exploit a web application via SSL. We account for the influence of encryption through random padding which is the most stringent condition in the protocol specifications. The 0-255 bytes of random padding were added to each size of transferred data to make the size multiple of the block size for the encryption algorithm.

The evaluation adopts 2 datasets, the actual dataset gathered at a network gateway and the DARPA IDS evaluation dataset (Fig. 5). The actual dataset consists of accesses to external web sites and attacks collected by a honey pot (Fawcett, 2006). It is obvious that the accesses from the LAN are legitimate activities and any accesses to the honey pot are malicious. Note that accesses from a LAN are TCP connections initiated from IP addresses that belong to a private address. We chose a dynamic web site to evaluate the system, as this site provides a social networking service and possesses functions to submit an study and comment on the study. Usually, attackers target these functions of a web application which are implemented as CGI and provide dynamic content (Gordon *et al*., 2005). The DARPA IDS evaluation dataset consists of PCAP formatted files that represent 5 weeks of traffic. We chose the files for weeks 4 and 5 because the detailed list describes attack instances that are restricted to weeks 4 and 5. While the attacks occurred against several protocols, we only use the HTTP attacks and any attacks observed on port 80 (Ingham, 2007). Table 1 shows details of the datasets. Request means the number of HTTP requests and an instance means the number of activities that the feature vector extraction outputs (Ingham and Inoue, 2007). The graph is generated by the

Table 1: Attacks in data sets

Attacks	Description
URL interpretation attack	Defining Wrong URL in the log server
Session hijacking	Defining the output as the time out period of the specified URL
Input validation attack	The input is not related to the url, so the validation error is occurred
Buffer overflow attack	Using error messages rejected by the database to find useful data facilitating injected of the backend database
SQL injection attack	SQL injection codes are injected into one or more conditional statement so that they are always evaluated to be true

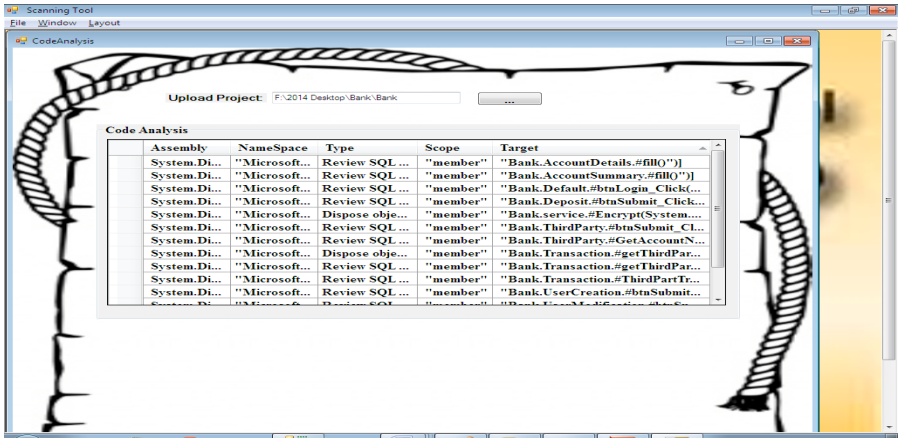

Fig. 5: Distribution of data's based on web attacks

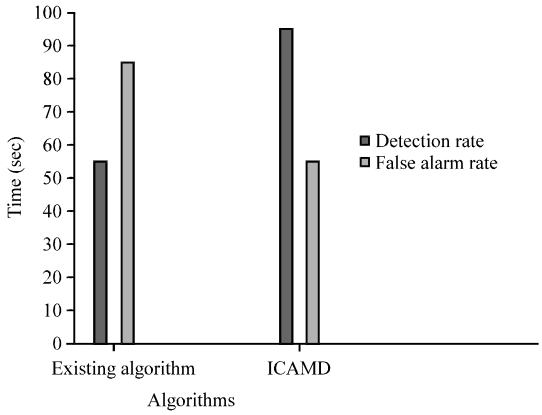

Fig. 6: Graphical representation on detection and false alarm rate of web attacks

values calculated by the threshold values. The main objective is to analyse the detection and false alarm rate.

Figure 6 explains the actual dataset; the Proposed algorithm detects the attacks with low false positive and false negatives rates. Actual instances of scanning, script and buffer overflow attacks are successfully distinguished from legitimate accesses with a high degree of accuracy (Luotonen, 1995; Maxion and Roberts, 2004). The Proposed algorithm captures the characteristics of an error response that the web server sends, namely, a small-sized response and the large-sized response. It means that the system cannot detect attacks that are similar to legitimate accesses. In the case of the DARPA

	Attributes					Class label
Instances	A_1	A_2	A_3	---	A_{41}	B
	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	---	$x_{1,41}$	y_1
	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	---	$x_{2,41}$	y_2
	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	---	$x_{3,41}$	y_3
	⋮	⋮	⋮	⋮	⋮	⋮
	$x_{n,1}$	$x_{n,2}$	$x_{n,3}$	---	$x_{n,41}$	y_n

Fig. 7: Vectors of each attributes based on the class labels

dataset, the attacker sends small-sized exploit code and gets a password file from the web server without any other scanning activity (Fig. 7).

However, in a typical scenario, an attacker would use scanning activities to carry out the attacks (Wang *et al*., 2006; Wang and Stolfo, 2004). To find an unknown vulnerability in a web application server, the attacker needs to try several scanning requests. The proposed algorithm would reveal unknown attacks by the related scanning activities, even if the unknown attacks are similar to legitimate accesses. Under such conditions, the approach would not be susceptible to high false positive/negative rates. In future research, we plan to evaluate the system using other traffic datasets and attack instances collected from actual networks.

CONCLUSION

Web intrusion detection is a promising technique since it allows detecting previously unknown attacks

which is important as new vulnerabilities and attacks are constantly appearing. The study presented a study of web based intrusion detection with a large web application. It presents how data was obtained and sanitized. A comparison of the several models that can be used to represent normal behaviour has shown that n-attacks with k sample conditions provide the best accuracy with a high detection rate and a low false positive rate.

REFERENCES

Cho, S. and S. Cha, 2004. SAD: Web session anomaly detection based on parameter estimation. Comput. Security, 23: 312-319.

Damashek, M., 1995. Gauging similarity with n-grams: Language independent categorization of text. Science, 267: 843-848.

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recog. Lett., 27: 861-874.

Gordon, L.A., M.P. Loeb, W. Lucyshyn and R. Richardson, 2005. CSI/FBI computer crime and security survey. Computer Security Institute,

Ingham, K.L. and H. Inoue, 2007. Comparing anomaly detection techniques for HTTP. Proceedings of the 10th International Conference on Recent Advances in Intrusion Detection, September 5-7, 2007, Springer-Verlag, Berlin, pp: 42-62.

Ingham, K.L., 2007. Anomaly detection for HTTP intrusion detection: Algorithm comparisons and the effect of generalization on accuracy. Ph.D. Thesis, University of New Mexico, USA.

Ingham, K.L., A. Somayaji, J. Burge and S. Forrest, 2007. Learning DFA representations of HTTP for protecting web applications. Comput. Networks, 51: 1239-1255.

Kruegel, C. and G. Vigna, 2003. Anomaly detection of webbased attacks. Proceedings of the 10th ACM Conference on Computer and Communications Security, October 27-31, 2003, Washington, DC, USA., pp: 251-261.

Kruegel, C., G. Vigna and W. Robertson, 2005. A multi-model approach to the detection of web-based attacks. Comput. Networks, 48: 717-738.

Luotonen, A., 1995. The common logfile format. July 1995. http://www.w3.org/Daemon/User/Config/Logging.html.

Maxion, R.A. and R.R. Roberts, 2004. Proper use of ROC curves in intrusion/anomaly detection. Technical Report CS-TR-871, Newcastle University, 2004.

Robertson, W., G. Vigna, C. Kruegel and R.A. Kemmerer, 2006. Using generalization and characterization techniques in the anomaly-based detection of web attacks. Proceedings of the 13th Symposium on Network and Distributed System Security, February 2006, San Diego, CA., pp: 1-15.

Roesch, M., 1999. Snort-lightweight intrusion detection for networks. Proceedings of the 13th LISA Conference on System Administration, November 7-12, 1999, Seattle, Washington, pp: 229-238.

Wang, K. and S.J. Stolfo, 2004. Anomalous payload-based network intrusion detection. Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection, September 15-17, 2004, Sophia Antipolis, France, pp: 203-222.

Wang, K., J.J. Parekh and S.J. Stolfo, 2006. Anagram: A content anomaly detector resistant to mimicry attack. Proceedings of the 9th International Conference on Recent Advances in Intrusion Detection, September 20-22, 2006, Springer-Verlag Berlin, pp: 226-248.