A Study on Anomaly Detection for Keystroke Biometrics Using Classification Technique

Mahesh S Nayak*, Dr. M. Hanumanthappa[1], Dr. S.Kavitha[2]

**Research and Development Centre,Bharathiar University,,Coimbatore – 641 046,INDIA*

mnayak67@yahoo.com

1 Professor, Department of Computer Science & Applications,Bangalore University, Bangalore560001, INDIA

hanu6572@hotmail.com

2 Assistant Professor, Dayananda Sagar College, Bangalore-560078

s.kavitha527@gmail.com

ABSTRACT

Anomaly detection is having prominent role in the current era. It is mainly used in the identification of data points, items, observations or events that do not conform to the expected pattern of a given group,as it ends in threat such as cyber intrusions or fraud. The anamoly detection will translate to the problem like a structural defect, medical problems or errors in a text or fraud in banking systems. In our research paper we have discussed about the various stages of data processing such as preprocessing to the classification. We have also compared the results obtained from the various algorithms to choose the best method to use for anomaly detection. K-Means clustering algorithm given the best result in 0.02 seconds time complexity.

1. Introduction

Anomalies generally known as noise, outliers,expectations or deviations. It is mainly applied in business area of research, finding strange patterns in network traffic which causes a data hack, health monitoring, fraud detection etc. The anamoly detection is classified as collective, contextual and point anamoly.In the first case, if one person is copying the information from the remote machine is meant to collective. In the second case of anamoly detection, time-series data of network will be used.In the point anamoly, some data will be guessed from inputs given.

Weka is a collection of **machine learning** algorithms for data mining tasks. It is having the algorithms for data pre-processing, classification, regression, clustering, association rules, and visualization.

2. Literature Survey

Anomaly detection has been the topic of a number of surveys and review articles, as well as books. Hodge and Austin [1] provide an extensive survey of anomaly detection techniques developed in machine learning and statistical domains. A broad review of anomaly detection techniques for numeric as well as symbolic data is presented by Agyemang et al. [2]. An extensive review of novelty detection techniques using neural networks and statistical approaches has been presented in Markou and Singh [6] and Markou and Singh [2003b], respectively.In general the academic literature on anamoly detection is sparse in literature. Honeynets [3] or darknets research paper distributed [4] or local is given data about anamoly detection using botnet field. The article [5] presents a presents the study of anamoly using botnet architecture.

3. Data Set Description

In the proposed research, Keystroke Dynamics - Benchmark Data Set is used for data clasiification technique. The bio.csv file is used and it is having 20400 instances and 34 attributes.The data set is executed with 'n' iterations and 15 fold using data mining tool.

4. Results and Discussions

In the paper we have used cfssubset algorithm for preprocessing[7]. This algorithm is a supervised learning algorithm. . The algorithm is more efficient as it evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them.

In this algorithm, features subsets correlated with the main class when low intensity values are incurred[8].Visualization of all the 42 attributes are shown in the Figure 1 and result of preprocessing is done in Figure 2.

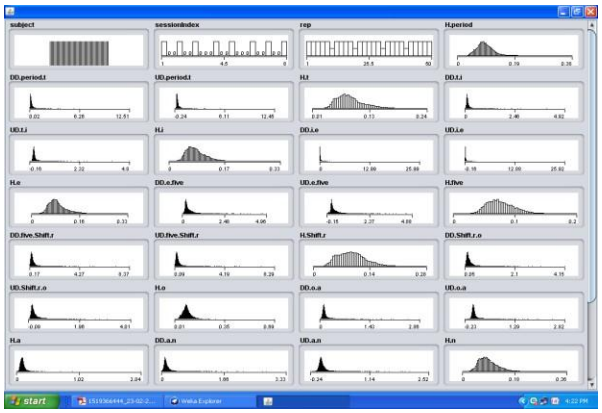

Figure 1. Visualization of keystroke biometrics

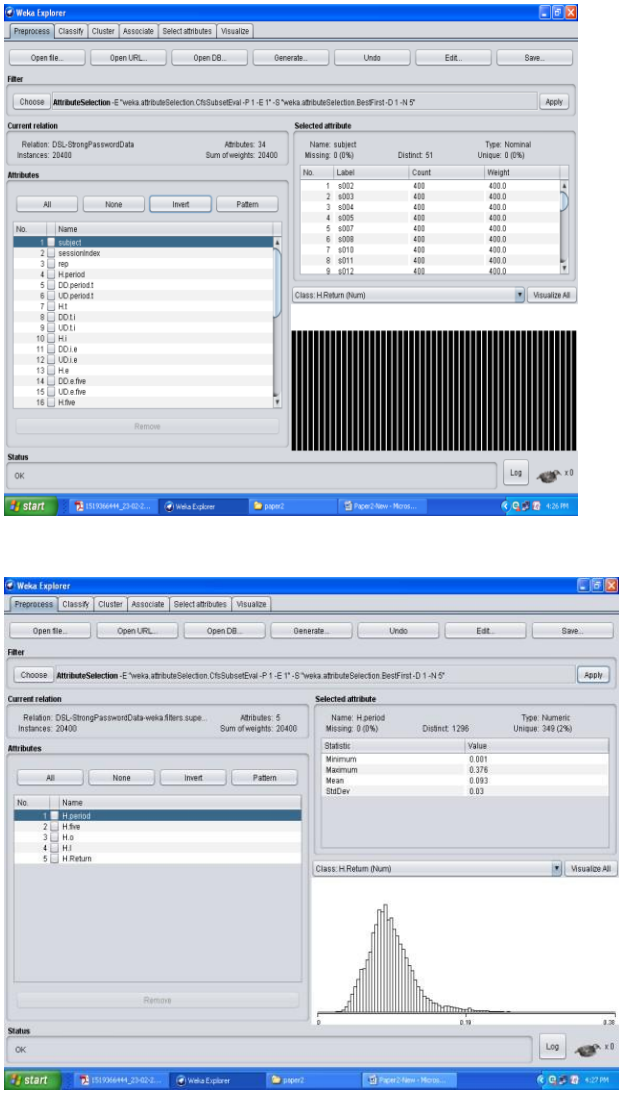

Figure 2. Preprocessing - the cfssubset evaluation algoithm.

4.1. Clustering

Canopy cluster

At every data, if its distance from the first point is < Data point(D1), then add the point to the cluster and if ,if it is less tha the second point D2, then delete the point from the data set. The canopy cluster process repeats until the initial set is finished,having a group of canopies. Canopy cluster is efficient algorithm in which is a fast and surprisingly accurate method for grouping objects into clusters. In the canopy cluster, for the processing the data set applies a fast approximate distance metric and two distance thresholds D1 > D2. And it is used to start with the data point for begin and remove one at random[9].The visualization of canopy cluster with all the attributes are shown in the Figure 3 and Figure 4 shows the Fathersfirst cluster.

Figure 3. Result of canopy cluster

4.2. Fathersfirst cluster

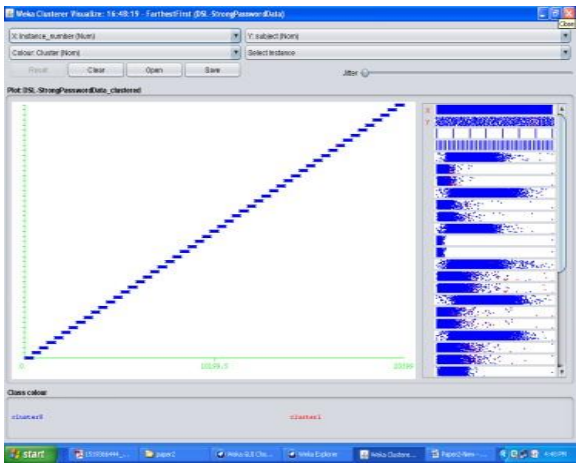

Figure 4. Result of Fathersfirst cluster

Number of iterations: 7

Within cluster sum of squared errors: 24337.172280706247

Initial starting points (random):

Time taken to build model (full training data) : 0.75 seconds

=== Model and evaluation on training set ===

Clustered Instances

0 10200 (50%)

1 10200 (50%)

Figure 5 shows the Filtered cluster graph visualization and the Figure 6 shows the Zero R algorithm for the given dataset.

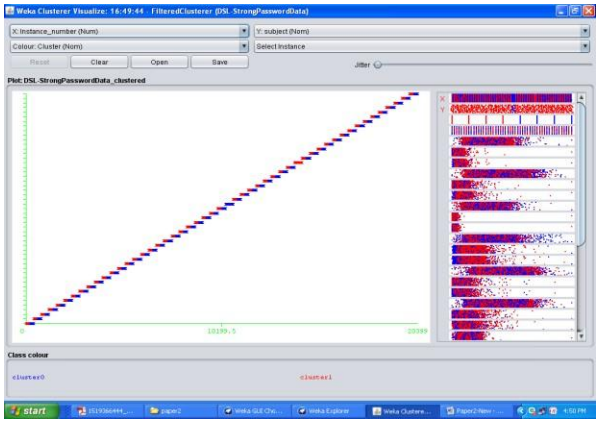

Figure 5. Filtered Cluster visualization

4.3 .kMeans cluster

Number of iterations: 7

Within cluster sum of squared errors: 24337.172280706247

Attribute	Full Data	Cluster# 0	1
	(20400.0)	(10200.0)	(10200.0)
subject	s002	s032	s041
sessionIndex	4.5	6.4216	2.5784
rep	25.5	25.5	25.5
H.period	0.0934	0.0923	0.0945
UD.period.t	0.1708	0.1385	0.203
H.t	0.0857	0.0867	0.0848
DD.t.i	0.1691	0.1603	0.1778
UD.t.i	0.0834	0.0737	0.093
H.i	0.0816	0.0822	0.081
DD.i.e	0.1594	0.1435	0.1752
UD.i.e	0.0778	0.0613	0.0943
H.e	0.0891	0.0904	0.0879
DD.e.five	0.3774	0.3136	0.4413
UD.e.five	0.2883	0.2232	0.3534
H.five	0.0769	0.0772	0.0766
DD.five.Shift.r	0.4389	0.3994	0.4784
UD.five.Shift.r	0.362	0.3222	0.4017
H.Shift.r	0.0959	0.0967	0.0952

DD.Shift.r.o 0.2509 0.2323 0.2695

UD.Shift.r.o 0.155 0.1357 0.1743

H.o 0.0884 0.0889 0.0878

DD.o.a 0.1569 0.1485 0.1654

UD.o.a 0.0686 0.0596 0.0776

H.a 0.1063 0.1056 0.1069

DD.a.n 0.1507 0.1385 0.1629

UD.a.n 0.0444 0.0328 0.056

H.n 0.0899 0.0915 0.0883

DD.n.l 0.2026 0.1854 0.2199

UD.n.l 0.1127 0.0939 0.1316

H.l 0.0956 0.0958 0.0953

DD.l.Return 0.3218 0.2915 0.3522

UD.l.Return 0.2263 0.1957 0.2569

H.Return 0.0883 0.0876 0.089

Time taken to build model (full training data) : 0.89 seconds

Clustered Instances

0 14035 (69%)

1 6365 (31%)

Log likelihood: 22.20447

4.4 Zero-R visualization and Running data

Time taken to build model: 0.02 seconds

Correlation coefficient -0.0205

Mean absolute error 0.0212

Root mean squared error 0.0275

Relative absolute error 100 %

Root relative squared error 100 %

Total Number of Instances 20400

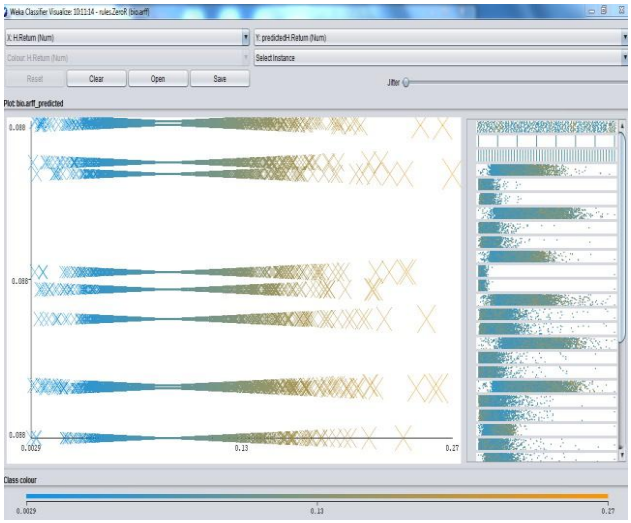

Figure 6. Visualization of Zero-R algorithm for anamoly detection

4.5 Decision table running information

Figure 7 shows the running information visualization of decision Table clustering algorithm.

Number of training instances: 20400

Number of Rules : 408

Non matches covered by Majority class.

Best first.

Start set: no attributes

Search direction: forward

Stale search after 5 node expansions

Total number of subsets evaluated: 212

Merit of best subset found: 0.016

Time taken to build model: 6.84 seconds

Correlation coefficient	0.8044
Mean absolute error	0.0113
Root mean squared error	0.0163
Relative absolute error	53.3996 %
Root relative squared error	59.4163 %
Total Number of Instances	20400

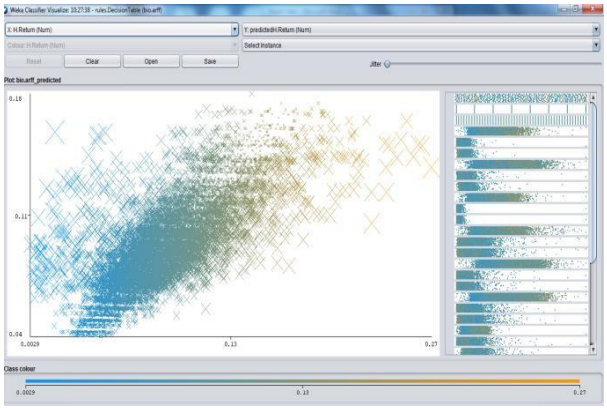

Figure 7. **visualization of Decision Table algorithm for anamoly detection**

Conclusions

Anamoly detection is having prominent role in big data analytics. It is having applications like behavioral analysis in order to aid in learning about the detection, identification and prediction of the occurrence of these anomalies. Even though many research is going in anamoly detection, in our research paper, have given the classification algorithm which is used less time complexity. The K- Means cluster algorithm can be used for any anomaly detection to speed up the process as it took only 0.02 seconds to build the model.

References

[1]. Hodge, V. and Austin, J. 2004. A survey of outlier detection methodologies. Artificial Intelligence Review 22, 2, 85–126

[2]. Agyemang, M., Barker, K., and Alhajj, R. 2006. A comprehensive survey of numeric and symbolic outlier mining techniques. Intelligent Data Analysis 10, 6, 521–538.

[3]. Markou, M. and Singh, S. 2003a. Novelty detection: a review-part 1: statistical approaches. Signal Processing 83, 12, 2481–2497

[4] M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson, The Internet Motion Sensor: A Distributed Blackhole Monitoring System. *In Proceedings of the Network and Distributed Security Symposium*, San Diego, CA, January 2005.

[5] P. Barford, V. Yegneswaran, An Inside Look at Botnets, *Special Workshop on Malware Detection, Advances in Information Security*, Springer Verlag, 2006

[6] The Honeynet Project and Research Alliance. Know Your Enemy, Tracking Botnets. http://honeynet.org/papers/bots, March 2005.

[7] Snort IDS web page. http://www.snort.org, March 2006

[8] Kevin S. Killourhy and Roy A. Maxion. "Comparing Anomaly Detectors for Keystroke Dynamics," in Proceedings of the 39th Annual International Conference on Dependable Systems and Networks (DSN-2009), pages 125-134, Estoril, Lisbon, Portugal, June 29-July 2, 2009. IEEE Computer Society Press, Los Alamitos, California, 2009. (pdf)

[9] T. Sing, O. Sander, N. Beerenwinkel, T. Lengauer. "ROCR: visualizing classifier performance in R," *Bioinformatics* 21(20):3940-3941 (2005). (link)