

# Heuristic Framework for Network Attack Detection Using Fuzzy ANN and Apache Mahout

Ronak Agrawal<sup>1</sup>, Ganesh Talekar<sup>2</sup>, Akshaysingh Chandel<sup>3</sup>, Shriganesh Munde<sup>4</sup>

, D. S. Zingade<sup>5</sup>

<sup>1</sup>BE, Computer, AISSMS IOIT, Pune, India

<sup>2</sup>BE, Computer, AISSMS IOIT, Pune, India

<sup>3</sup>BE, Computer, AISSMS IOIT, Pune, India

<sup>4</sup>BE, Computer, AISSMS IOIT, Pune, India

<sup>5</sup>ME, Computer, AISSMS IOIT, Pune, India

## ABSTRACT

As in today's world the network has become more vulnerable to the attacks so there is a need of making the network more secure and our project is an initiative process towards it. In our project we have computed the probabilities of the attacks and intrusions which have been detected. In market there are plenty of different IDS available but none of the distributed type. Many of them make use of pattern matching and different complex algorithms but this makes the system slow. So for proper intrusion detection in distributed network we are making use of the concepts like K-means clustering, Fuzzy, Artificial neural network, Bayesian Learning, Apache Mahout.

**Keyword:** - K-means, Artificial Neural Network, Fuzzy, Bayesian Network, Apache Mahout.

## 1. INTRODUCTION

For the detection of the intrusions in the distributed network we made use of various concepts like k-means clustering, Artificial neural network, Fuzzy logic, Bayesian learning and Apache Mahout. Traditional ways of processing and analysing of critical traffic in networks may not be efficient in practice. In this work, we focus on building an efficient network management system.

We will be taking real-time network dataset as input for all the computation. These data is initially in the csv format. These data will be then distributed into the different no of clusters by making use of the k-means clustering algorithm for the efficient analysis of the data. The clusters are scalable and making use of more cluster the results obtain is more precise. After that we will make use of ANN, Fuzzy logic, Bayesian learning and Apache Mahout for the further processing and analyzing the intrusions.

The Apache Hadoop system is an important system for handling massive volumes of data. Distributed network measurement system can be implemented using a best effort in parallel and distributed Machine Learning (Mahout). Apache Mahout is used for the purpose of the collaborative filtering of the probabilities obtained by the Bayesian learning algorithm. However, it is not able to support real-time analysis due to Mahout's limitation, as it is built on top of the Hadoop HDFS file. The rest of the paper is organized as follows. In Section II, we present our framework and explain various components. Preliminary study results are presented in Section III. Finally, we conclude with a summary in Section IV.

## 2. PROPOSED METHODOLOGY

This section describes the steps that are been taken to remove haze from the images with the below mentioned steps as depicted in figure 1.

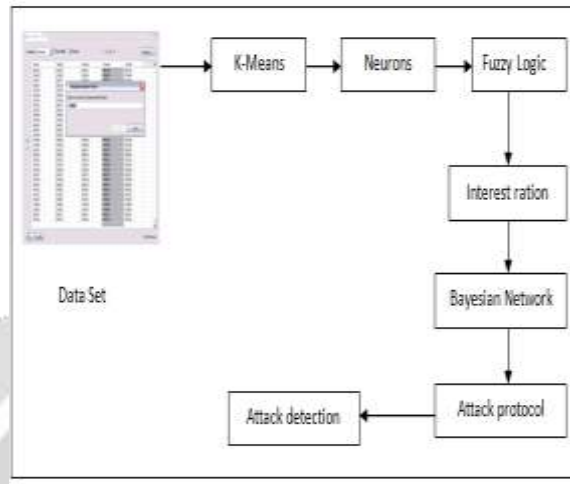


Figure 1: System overview

**Step 1:** In this starting step the dataset is being selected by the user. The dataset is of networking, it contains various parameters of the networks such as source ip, destination ip, port numbers etc.

**Step 2:** Once the dataset is being selected by step 1 all the necessary parameters of the dataset are stored in a vector which will be used in further operations.

**Step 3:** this step is known as a grouping step as the data is being grouped to the n groups also known as cluster, where n is a user defined number. To accomplish the grouping task k means algorithm by James MacQueen developed in 1967 is used. K means is the well-known and widely used clustering algorithm. K means clusters the data in such way that the data belonging to one group has a similar nature. Here Euclidean distance is calculated using different attributes and this distance is then used for the k means algorithm.

**Step 4:** As the k mean clustering is unsupervised and abstract clustering it fails to group the data accurately. So we imposed artificial neural network for the fine classification. ANN creates the neurons for each of the clusters. For each cluster two neurons are created, so if the cluster number is 5 then there will be the 10 neurons.

**Step 5:** Here in this step fuzzy logic is introduced. Fuzzy inference engine creates the five ranges which are being used while attack detection.

**Step 6:** Once the fuzzy ranges are calculated interest ratio is calculated. For calculation of interest ratio destination ip is fetched and all the associated source ip are calculated. This calculation is done on the neurons obtained in last step. So it gets easy to display it according to neurons. The motto behind the use of interest ration is to calculate the incoming ip's.

**Step 7:** the interest ratio output is then fed to the Bayesian probability calculation module. Here in this step probability of each of the destination ip is calculated. Bayesian probability is well narrated in diagram below.

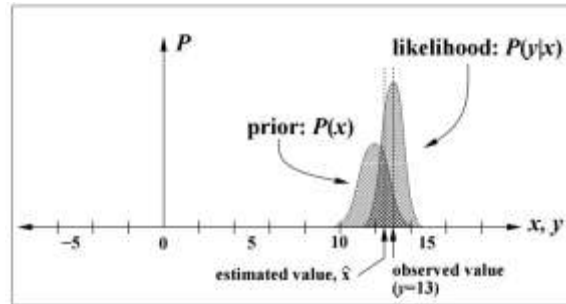


Figure 2: Bayesian probability

Bayesian probability is well calculated using formula below.

$$\frac{p\left(\frac{B}{A}\right) p(A)}{p(B)}$$

**Step 8:** Here in this phase a filter is applied on calculated Bayesian probabilities. All the Bayesian probabilities are sorted according to occurrence of destination ip in the original dataset.

**Step 9:** in this step filtered probability is optimized .All the probabilities exceeding 1 are restricted to 1.

**Step 10:** Here attack is identified. The occurrence of each of the destination ip is fetched, the destination ip whose occurrence is greater than the high ranges of the fuzzy set in step 5 are fed as a input to the attack detection.

**Step 11:** We are also using Apache Mahout for the collaborative filtering of the attacks detected.

**Step 12:** Using the Fuzzy Logic Protocols set in step 5 Attacks like DOS and D-DOS are identified.

The complete process Neuro fuzzy technique can n be depicted with the following algorithm

**Input:** Complete dataset  $D_s$  and no of clusters  $N$

**Output:** clusters and neurons

**Step 0:** Start

**Step 1:** Read all the attributes of dataset  $D_s$

**Step 2:** Find Euclidean distance for each of the dataset row.

**Step 3:** find smallest and biggest Euclidean distance and fed it to the fuzzy logic.

**Step 4:** Generate  $N$  clusters from  $D_s$  using ranges generated in step 3

**Step 5:** Find sd, mean and Gaussian function of each cluster by considering two important attributes i.e.  $d_1, d_2$

**Step 6:** find minimum range and maximum range of each clusters for two attributes

**Step 7:** minimum range =mean

If (Gaussian value > (mean \* 2))

Maximum range=mean + sd

Else

Maximum range =mean + Gaussian function

**Step 8:** Apply ANN on  $C_N$  to generate neurons by using minimum range and maximum range

**Step 9:** store all the newly generated neurons to  $N_C$

**Step 10:** return clusters  $C_N$  and neurons  $N_C$

**Step 11:** Stop

### 3. RESULTS AND DISCUSSIONS

To show the effectiveness of proposed system some experiments are conducted on java based windows machine. To measure the performance of the system we set the bench mark by selecting network dataset which is collected over the live router's end in MS excel format and which contain all the fields of the data packets in the network.

To determine the performance of the system, we examined how many relevant data packets are considered as the threat generating to the network situation awareness based on adoptive grey verhulst model.

To measure this precision and recall are the best measuring techniques. So precision can be defined as the ratio of the number of relevant threats predicted to the total number of irrelevant and relevant threats predicted. It is usually expressed as a percentage. This gives the information about the relative effectiveness of the system.

Whereas Recall is the ratio of the number of relevant threats predicted to the total number of relevant threats predicted. It is usually expressed as a percentage. This gives the information about the absolute accuracy of the system.

The advantage of having the two for measures like precision and recall is that one is more important than the other in many circumstances. In contrast, various professional searchers and intelligence analysts are very concerned with trying to get as high recall as possible, and will tolerate fairly low precision results in order to get it. Individuals searching their hard disks are also often interested in high recall searches. Nevertheless, the two quantities clearly trade off against one another.

For more clarity let we assign

- A = The number of relevant threats Selected,
- B = The number of relevant threats predicted, and
- C = The number of irrelevant threats predicted.

So, Precision =  $(A / (A + C)) * 100$

And Recall =  $(A / (A + B)) * 100$

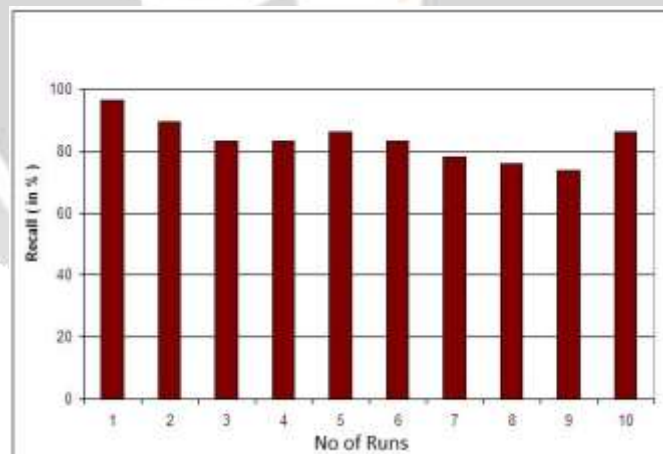


Figure 3: Average precision of the proposed approach

In Fig. 3, we observe that the tendency of average precision for the relevant threats predicted are high compared to other systems.

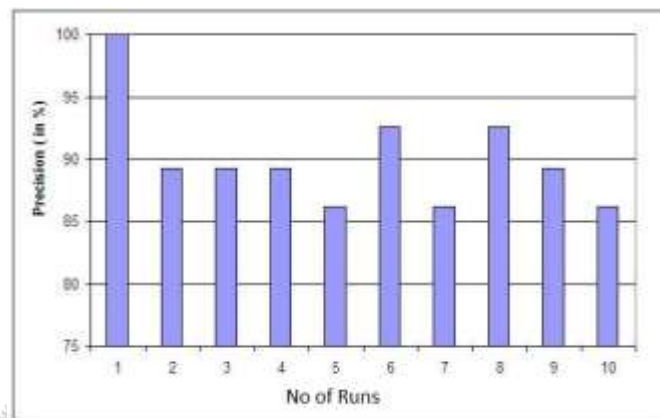


Figure 4: Average Recall of the proposed approach

In Fig. 4, we observe that the tendency of average Recall for the relevant threats predicted are high compared to other system. So this shows that our proposed system is achieving high accuracy than any other method.

#### 4. CONCLUSIONS

In this preliminary work, we present an idea of proper Distributed intrusion detection system using Bayesian learning in Apache mahout using key-concepts like K-means clustering, Fuzzy, Artificial neural network, Bayesian Learning, Apache Mahout. We obtained preliminary results for distributed network anomaly detection. Future research will be on in-depth analysis of anomaly detection for a distributed network management system, enhancement of machine learning and optimization algorithms for realtime processing and high accuracy, and implementation of visualizing tools for comprehensive understanding of dynamic behaviors of complex networks.

#### 5. ACKNOWLEDGEMENT





It gives us great pleasure in presenting the paper on 'Distributed Intrusion Detection System Using Bayesian Learning and Apache Mahout'. I would like to take this opportunity to thank Prof. D.S.Zingade for giving us all the help and guidance we needed. In the end our special thanks to Prof. S.P.Pimpalkar, Prof. Amol Kalugade, Prof. S.N.Zaware for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project. I am really grateful to them for their kind support and their valuable suggestions were very helpful.

#### 6. REFERENCES

- [1] Algorithms James, W. Myers Kathryn, Blackmond Laskey and Tod S. Levitt, "Learning Bayesian Networks from Incomplete Data with Stochastic Search Algorithms Learning Bayesian Networks from Incomplete Data with Stochastic Search", Society for industrial & applied mathematics (SIAM), 2006.
- [2] Yoshinori Tamada, Seiya Imoto and Satoru Miyano "Parallel Algorithm for Learning Optimal Bayesian Network Structure", ACM, 2011.
- [3] Cassio P. de Campos, Qiang Ji Olga Nikolova, Jaroslaw Zola, and Srinivas Aluru "Efficient Structure Learning of Bayesian Networks using Constraints", ACM, 2011.
- [4] Cassio P. de Campos, Qiang Ji Olga, Nikolova, Jaroslaw Zola, and Srinivas Aluru "A Parallel Algorithm for Exact Structure Learning of Bayesian Networks", IOWA State university, 2012.

- [5] Timo, J. T. Koski (Stockholm) and John M. Noble (Warsaw), "A Review of Bayesian Networks and Structure Learning", *Mathematica Applicanda*, 2012.
- [6] Tommi Jaakkola, David Sontag, Amir Globerson and Marina Meila "Learning Bayesian Network Structure using LP Relaxations", *NYU Computer science*, 2010.
- [7] Cassio P. de Campos, Zhi Zeng and Qiang Ji, "Structure Learning of Bayesian Networks using Constraints", *ACM*, 2009.
- [8] Ahmed Mabrouk, Christophe Gonzales, Karine Jabet-Chevalier and Eric Chojnaki, "An Efficient Bayesian Network Structure Learning Algorithm in the Presence of Deterministic Relations", *DESIR(Paris)*, 2014.
- [9] Harald Steck, "Learning the Bayesian Network Structure: Dirichlet Prior versus Data", *Cornell University Library*, 2012.
- [10] Alexandru Niculescu-Mizil and Rich Caruana, "Inductive Transfer for Bayesian Network Structure Learning", *Springer*, 2012.
- [11] Carol J Fung, Jie Zhang and Raouf Boutaba, "Effective Acquaintance Management based on Bayesian Learning for Distributed Intrusion Detection Networks", *IEEE Transactions on network and service management*, VOL. 9, NO. 3, September 2012
- [12] Shuai Zhao, Mayanka Chandrashekar, Yugyung Lee, Deep Medhi, "Real-Time Network Anomaly Detection System Using Machine Learning", *International Conference on the Design of Reliable Communication Networks (DRCN)*, 2015.
- [13] Apache Mahout <https://mahout.apache.org>

## BIOGRAPHIES

	<b>Ronak Agrawal</b> , pursuing B.E Computer Engineering in Savitribai Phule Pune University. He is also interested in Computer Networking, Apache Hadoop and Operating System.
	<b>Ganesh Talekar</b> , pursuing B.E Computer Engineering in Savitribai Phule Pune University. He is also interested in Database Management System and Android Based Application.
	<b>Akshaysingh Chandel</b> , pursuing B.E Computer Engineering in Savitribai Phule Pune University. He is also interested in Embedded Operating System.
	<b>Shriganesh Munde</b> , pursuing B.E Computer Engineering in Savitribai Phule Pune University. He is also interested in Parallel & Distributed Computing and Data Mining.



**Deeplakshmi Zingade**, completed her M.E from Savitribai Phule Pune University in 2013. She received her B.E from Rashtrasant Tukadoji Maharaj Nagpur University in 2003. She is currently working as Assistant Professor in AISSMS IOIT Pune

