Network Intrusion Detection Using Improved Genetic k-means Algorithm

Anand Sukumar J V, Pranav I, Neetish MM, Jayasree Narayanan
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham,
Amritapuri
India
Email: anandsukumar96@gmail.com, pranavtirur1@gmail.com,
neetishmahesh96@gmail.com, jayasreen@am.amrita.edu

***Abstract*—Internet is a widely used platform nowadays by people across the globe. This has led to the advancement in science and technology. Many surveys show that network intrusion has registered a consistent increase and lead to personal privacy theft and has become a major platform for attack in the recent years. Network intrusion is any unauthorized activity on a computer network. Hence there is a need to develop an effective intrusion detection system. In this paper we acquaint an intrusion detection system that uses improved genetic k-means algorithm(IGKM) to detect the type of intrusion. This paper also shows a comparison between an intrusion detection system that uses the k-means++ algorithm and an intrusion detection system that uses IGKM algorithm while using smaller subset of kdd-99 dataset with thousand instances and the KDD-99 dataset. The experiment shows that the intrusion detection that uses IGKM algorithm is more accurate when compared to k-means++ algorithm.**

***Keywords*—Internet, Network Intrusion, Intrusion Detection, Intrusion Detection System, IGKM algorithm, k-means++ algorithm, KDD-99.**

I. Introduction

People all over the world use the world wide web to such an extent that the Internet has become a crucial part in their lives. People use the Internet for many purposes such as involvement in social media, currency transactions, exchange of personal information and also storing private data such as passwords, personal media, banking details like credit card credentials. The world wide web has advanced to such an extent that it has developed from a set of markup language sites to a place where performing remote actions on a network from any where in the world is an easy task. Surveys show that network intrusion crimes have increased drastically over the years and lead to personal privacy theft. The data that is stolen as part of personal privacy theft is sold in black markets. Hence there is a need to develop an effective and efficient network intrusion detection system for detecting the type of attacks.

In this paper we present an intrusion detection system that uses improved genetic k-means algorithm to detect the type of intrusion. The system uses clustering to detect the type of spams. Clustering is a type of unsupervised learning that involves the partitioning of a set of data into a set of meaningful sub-classes called as clusters. Most cluster based intrusion detection systems use the traditional k-means algorithm to detect the type of attack. The drawback of the traditional k-means algorithm is that the value of 'k' (i.e, the number of clusters) must be set beforehand. This method is easy because small datasets are used but it is tedious to fix the value of 'k' manually when larger datasets are used. Hence our project uses the improved genetic k-means algorithm. The improved genetic k-means algorithm uses a fitness function which helps in finding the optimal value of 'k' with high accuracy. Fitness function uses inner cluster distance as well as inter cluster distance. Hence the fitness function is a product of three factors. Maximizing the factor leads to generation of cluster values. Our project uses the KDD-99 cup dataset for training the IGKM algorithm. KDD-99 dataset is a consistent and widely used dataset in the field of network intrusion detection. It is the subset of DARPA-98 dataset. KDD-99 dataset is a multi-variate dataset. The dataset contains 4.8 million instances. The characteristics of the attributes used in the dataset is categorical and integer in nature. It has forty two attributes. The Dataset mainly consists of four types of attacks-

DOS : Denial of Service. eg:- syn flood
R2L : Unauthorized access from remote machines. eg:- guessing passwords
U2R : Unauthorized access to local superuser (''root'') privileges. eg:- buffer overflow attacks.
probing : Surveillance and other probing. eg:- port scanning

The functionality of the project is clustering the KDD-99 dataset and using clusters the generated to identify the type of attack. KDD-99 dataset is a huge dataset with 42 headers. If the dataset is taken as it is, for the input of the IGKM algorithm run time of the intrusion detection system increases. Hence to reduce the run time attribute reduction is performed on the entire dataset. After attribute reduction the dataset is reduced from 42 headers to 6 headers and hence a reduced KDD-99 dataset is generated. This dataset is used as input for training the IGKM algorithm.

During the training phase sixty percent of the dataset is given as input for the IGKM algorithm. It then generates the

978-1-5386-5314-2/18/$31.00 ©2018 IEEE

optimum value of 'k' and hence the entire training dataset is clustered. During the testing phase unknown input is given into the intrusion detection system, system then uses the clusters generated and predicts whether it is a spam or not.
Our project also shows a comparison between an intrusion detection system that uses a k-means++ algorithm and an intrusion detection system that uses an IGKM algorithm by means of line chart. The accuracy of IGKM algorithm is shown using precision and recall.

II. Introduction to Clustering algorithm used

In this section, k-means algorithm and IGKM algorithm are introduced briefly.

A. k-means Algorithm

This algorithm starts at the random point and iterates between the first order optimality conditions with respect to the set of variables Z_ij, W_ij. The algorithm stops when no further improvement is attainable. The details refer to reference[2].

```
N:  number of data objects
K:  number of clusters

objects[N]:  array of data objects
clusters[K]:  array of cluster centers
membership[N]:  array of object memberships


kmeans_clustering( )
1   while  δ/N > threshold
2      δ ← 0
3      for i ← 0 to N-1
4         for j ← 0 to K-1
5            distance ← | objects[i] - clusters[j] |
6            if distance < dmin
7               dmin ← distance
8               n ← j
9         if membership[i] ≠ n
10           δ ← δ + 1
11           membership[i] ← n
12        new_clusters[n] ← new_clusters[n] + objects[i]
13        new_cluster_size[n] ← new_cluster_size[n] + 1
14     for j ← 0 to K-1
15        clusters[j][*] ← new_clusters[j][*] / new_cluster_size[j]
16        new_clusters[j][*] ← 0
17        new_cluster_size[j] ← 0
```

Fig. 1: Traditional k-means Algorithm.

B. IGKM Algorithm

In IGKM algorithm the number of clusters is not known before-hand. The optimal value of k is found using GA(Genetic algorithm). The fitness function(evaluating function) keeps the number of clusters as small as possible and increasing the separation and effectiveness as much as possible. Through the fitness function we get the optimal value of k and the data is clustered till n-generations. Hence we get the optimal clusters. The details refer to reference[1]. steps of the algorithm can be abridged as follows:

Algorithm: Improved Genetic K-means (S,k), S = { X_1,X_2,.....,X_n}.
Input: The number of clusters K'(K'¿K) and a dataset containing n objects X_i.
Output: A set of k clusters C_j that minimize the squared-error criterion.

Begin
1. Multiple sub-samples {S_1, S_2, ...,S_j};
2. For m = 1 to j do
   Genetic K-means(S_m, K'); //executing Genetic K-means, produce K' clusters and j groups.
3. Compute$J_c(m) = \sum_{j=1}^{k'} \sum_{X_i \in C_j} |X_i - Z_j|^2$;
4. Choose min{J_c} as the refined initial points Z_j , j ∈ [1, K'];
5. Genetic K-means(S, K'); //executing Genetic K-means again with chosen initial, producing K'mediods.
6. Repeat
   Combining two near clusters into one cluster, and recalculate the new center generated by two centers merged.
7. Until the number of clusters reduces into k //Merging (K' +K)
End

Fig. 2: IGKM Algorithm.

III. IDS Design

The intrusion detection system design is shown in Fig.3. The intrusion detection system can be secluded into a the following factors:

A. Dataset

The dataset used in this intrusion detection system is KDD-99 cup dataset. The dataset consists of 42 attributes that show the characteristics of different data points in the dataset. This dataset contains 4.8 million instances. The dataset consists of four main types of intrusions namely dos, r2l, u2r and probing. The above said intrusion types can be further classified into 22 types of attacks. The details refer to reference[3]. This dataset is used to find the nature of IGKM algorithm in Larger datasets. In this paper we also use a smaller subset of the KDD-99 dataset with thousand instances. This dataset is used to find the nature of IGKM algorithm in smaller datasets.

B. Attribute Reduction

The KDD-99 dataset is a very large dataset with 42 attributes and well over 4.8 million instances. So while using large datasets the runtime of the intrusion detection system exceeds well beyond the feasible limit. Hence to make the intrusion system more optimized attribute reduction is performed on the data set. In attribute reduction dispensable attributes are removed from the knowledge(dataset) while

maintaining the knowledge consistency. The attributes are reduced by combining similar types of attributes. Addition or multiplication or reciprocal operations are performed on the dataset to reduce the dataset to a dataset with lesser number of headers.

C. Fitness function

Fitness function is used to find the optimal value of the number of clusters(k). In this project fitness function combination of two parts namely the inner cluster distance and inter cluster distance.

1) Inner cluster distance: The inner cluster distance is represented by E_k.

$$E_k = \sum_{j=1}^{k} \sum_{i \in I_j} \left\| x_i - z_j \right\|^2$$

Where k is the number of clusters and the inter cluster distance is calculated as the sum of the squares of difference between set of indices in the cluster *j* with the centroid of the cluster j(Z_j).

2) Inter cluster distance: The inter cluster distance is represented by D_k.

$$D_k = \max_{i,j=1}^{k} \left\| z_i - z_j \right\|^2$$

Inter cluster distance can be calculated as the maximum of the squares of difference between centroids(Z_i,Z_j) of clusters i and j. Fitness function can be calculated as

$$fitness(k) = \frac{1}{k} \times \frac{E_1}{E_k} \times D_k$$

Fitness function is explained in detail in reference[2].

D. Training phase

The process that takes place under the training phase is that we are making the algorithm learn by giving known inputs. In this phase the IGKM algorithm is trained with the attribute reduced dataset. The dataset is clustered and the number of clusters to be generated is the optimal value that is fostered from the fitness function. In the training phase the IGKM algorithm is trained with sixty percent of the KDD-99 dataset and clusters are generated. To get a more optimized clusters the number of generations is taken as ten.

E. Testing phase

The process that takes place while testing is the system is given unknown inputs and we check whether the output is correct or not. During this phase random input values from the remaining forty percent of the KDD-99 dataset is given as input. The system then uses the clusters generated during the training phase to check the type of attack.

F. Classifier

The intrusion detection system uses the classifier to check whether the output generated from the algorithm is correct or not. Our classifier uses ID-mapping to check the correctness of output. During attribute reduction the dataset is reduced to a seven attribute dataset, of which one of the attribute is ID number. In the official KDD-99 dataset the ID number shows the reference to the corresponding instance. The ID number of the output is used to cross reference and check for correctness.

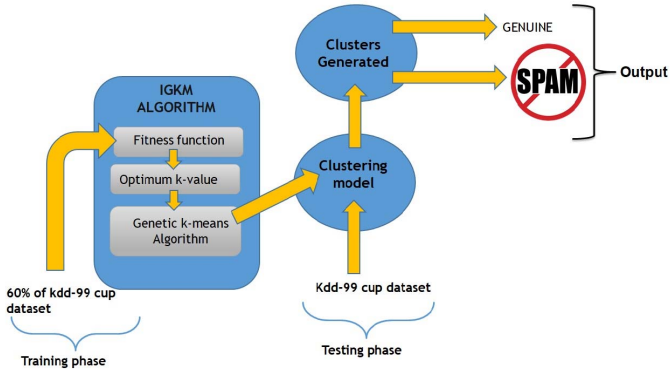

Fig. 3: Design of intrusion detection system that uses IGKM algorithm.

IV. Output Evaluation

The output evaluation is done using a series of factors-

- *True positive* (TP):This refers to the number of positive tuples that were correctly labeled by the classifier.
- *False positive* (FP):This refers to the number of negative tuples that were incorrectly labeled by the classifier.
- *False Negative* (FN):These are the positive tuples that were mislabeled as negatives.
- *Precision*: The ratio of number of true positives to the number of false positives.

$$Precision = (TP/FP)$$

- *Recall*: The ratio of number of true positives to the sum of number of false positives and false negatives.

$$Recall = TP/(FP + FN)$$

- *Accuracy* (ACC): This is the total accuracy of the classifier.

$$ACC = (Precision/Recall)$$

V. Comparison

In this paper we compare IGKM algorithm and k-means++ algorithm. First we use a smaller subset of KDD-99 dataset with thousand instances to simulate a smaller dataset. Then we use the KDD-99 dataset to simulate a larger dataset.
Firstly, we run both the algorithms on a smaller subset of KDD-99 dataset with thousand instances. The Prediction Values are shown in TABLE I. The accuracy of both the algorithms while using the smaller subset of KDD-99 dataset with thousand instances is shown in TABLE II.
Secondly, we run both the algorithms on KDD-99 dataset. The prediction values are shown in TABLE III. The accuracy of both the algorithms while using the KDD-99 dataset is shown in TABLE IV.

TABLE I: Comparison of TP, FP and FN values of k-means++ and IGKM algorithm for smaller subset of KDD-99 dataset with thousand instances.

Algorithm	TP	FP	FN
k-means++	3	599	43
IGKM	4	19	4

TABLE II: Comparison of accuracy of k-means++ and IGKM algorithm for smaller subset of KDD-99 dataset with thousand instances.

Algorithm	Precision	Recall	Accuracy
k-means++	0.005008	0.004673	53.271028%
IGKM	0.352941	0.260870	26.086957%

TABLE III: Comparison of TP, FP and FN values of k-means++ and IGKM algorithm for KDD-99 dataset.

Algorithm	TP	FP	FN
k-means++	3	539	103
IGKM	6	17	6

TABLE IV: Comparison of accuracy of k-means++ and IGKM algorithm for KDD-99 dataset.

Algorithm	Precision	Recall	Accuracy
k-means++	0.005566	0.004673	53.271028%
IGKM	0.352941	0.268869	72.913043%

VI. Results

While using smaller subset of KDD-99 dataset with thousand instances the k-means++ algorithm gives an accuracy of 53.27% and IGKM algorithm gives an accuracy of 26.08%. While using the k-means++ algorithm gives an accuracy of 53.27% and IGKM algorithm gives an accuracy of 72.91%. The accuracy of IGKM algorithm is higher for large datasets, this is because IGKM algorithm is used to cluster large datasets. Graphs in fig.4 and fig.6 show the nature of fitness for small and large datasets.
In the case of smaller datasets the time complexity of IGKM algorithm is higher when compared to the k-means++ algorithm as the IGKM algorithm uses a fitness function to determine the value of optimum number of clusters. This is shown in fig.5 .
In the case of larger datasets the time complexity of IGKM algorithm is lower when compared to the K-means++ algorithm as the fitness function used in the IGKM algorithm finds the optimal value of K prior to the execution of the clustering phase. There by increasing the accuracy of the clusters formed. This is shown in fig.7 .

Fig. 4: Graph of fitness for smaller subset of KDD-99 dataset with thousand instances.

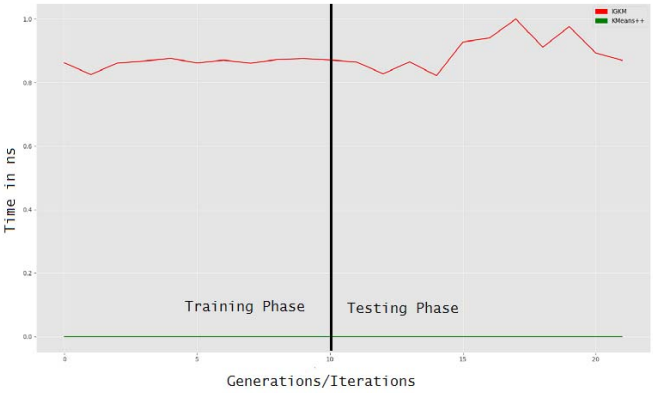

Fig. 5: Time complexity comparison between k-means++ and IGKM algorithm for smaller subset of KDD-99 dataset with thousand instances.

VII. Related Works

The studies have been mentioned in various publications. The activity of network intrusive features have been mentioned in reference[4]. It observes that there has been a tremendous increase in the number of attacks and it does a study on the intrusion detection methods and type of attacks. The goal of the system is defined to be as to detect the anomalies in its characteristics as well as misuse in networks. The paper observes intrusion detection systems a burglar alarm which is like the lock system in houses. The paper gives a special

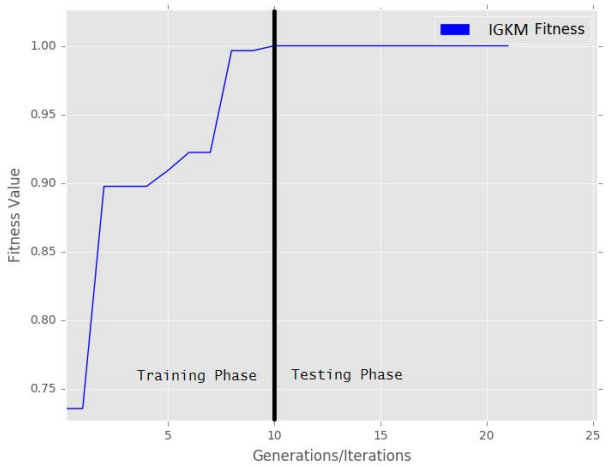

Fig. 6: Graph of fitness in KDD-99 dataset.

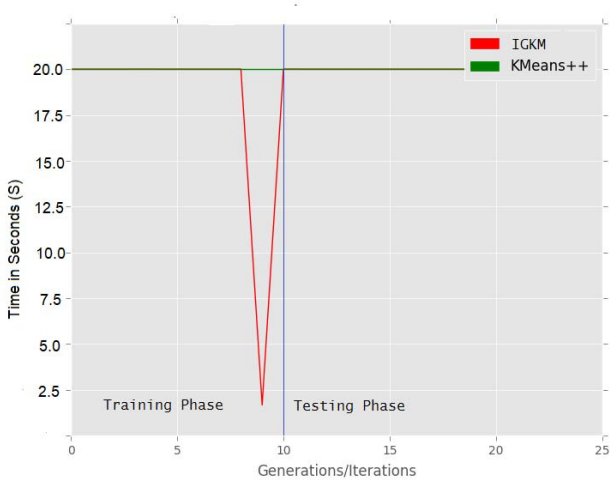

Fig. 7: Time complexity comparison between k-means++ and IGKM algorithm for KDD-99 dataset.

mention to the activities of firewall which is mentioned as a filtering mechanism for external connections and activities of those type of objects. The paper observes intrusion detection system as into three types - Host based IDS, Network based IDS and Application based IDS. It also describes the advantage of each of these systems. It contains the functionality of these systems as well. The functionality is focused on four types- Data collection, Feature selection, Analysis and Action.

In reference[3] it works on selection of important features which in turn leads to simplification of problem with a more accurate rate of detection. They have highlighted the relevance of each feature in KDD-99 dataset to that of detection in each classes. The need for such a system is triggered by the fact that protective measures such as firewall have failed to track the motion of intrusive objects. Selection of certain features in the learning phase is a crucial part in deciding the risk factor. This selection leads to decrementing the cost of computing as well as resizing the data and also the accuracy. The paper also presents the concept of rough set. The imperfection in precision and data as well as patterns that are being hidden can be traced with the activity of rough sets. It also studies a concept, Entropy which is the method to determine the output related with a subset on the basis of information delivered by an input. Mapping of labels in class and computation of degree of dependency on the basis of availability of instances of class are the methods used to find the significance of features in a class.

In reference[1] there is a comparison between the traditional k-means algorithm and GA-based algorithm, GKM algorithm and a new algorithm called improved genetic k-means algorithm. The improved genetic k-means algorithm tries to overcome the flaws faced by most of the other algorithms mentioned above. In the case of k-means algorithm the drawback is that the value of number of clusters must be fixed prior to running the algorithm. The above said flaw is not much of a concern in the case of small datasets. But it poses a major problem when large datasets are used. GA-based method when compared with the traditional k-means algorithm gives more optimized clusters as output. The third algorithm GKM is a combination of the first two algorithms and also contains the value of fitness which gives more accurate clusters. But this algorithm also faces the same drawback of traditional k-means algorithm. The new algorithm proposed in this paper overcomes all the drawbacks faced by algorithms mentioned above. As this algorithm uses a fitness function to determine the optimal value of k and also computes the inter cluster distance and inner cluster distance. These distance values are used to find the optimal centroids for generating clusters. Hence IGKM algorithm can cluster large datasets without even knowing the value of the number of clusters. Hence it gives more accurate clusters when compared to the other algorithms mentioned above.

In reference[5] they finds the flaws in the existing security systems like various anti-virus systems as well as firewall and IDS. Many initial events are being probed in order to cluster those which come under malicious as attack types. Various types of attacks are being studied and a more modified solution is found out on the basis of this.

VIII. Conclusion

Intrusion crimes is increasing day by day. Hence there is need to find the optimal intrusion detection system when compared to the intrusion detection systems that use the traditional clustering algorithms. In this paper, we developed an intrusion detection system that uses IGKM algorithm to detect the type of intrusion and the number of clusters (k) is not fixed beforehand. The optimal value of k is found using the fitness function which helps to effectively generate optimized clusters thereby contributing to the effective detection of the type of attack. From the experiments done in this paper we can conclude that intrusion detection system that uses IGKM algorithm shows lesser accuracy when smaller datasets are

used but, in the case of large datasets intrusion detection system that uses IGKM algorithm shows comparatively higher accuracy when compared to the intrusion detection systems that uses k-means clustering algorithm.

ACKNOWLEDGMENT

The authors wish to thank their mentors Mrs. Archana.K.Rajan and Mrs. Jayasree Narayanan and the Department of Computer Science for their helpful guidance, suggestions and for providing a better outlook regarding this project. We gratefully acknowledge Amrita Vishwa Vidyapeetham, Amritapuri, India for providing us a platform to develop this project. We also wish to thank the anonymous reviewers who helped in the successful completion of this project.

REFERENCES

[1] Hai-xiang Guo, Ke-jun Zhu, Si-wei Gao, Ting Liu, "An Improved Genetic k-means Algorithm for Optimal Clustering", *Sixth IEEE International Conference on Data Mining (2006)*.

[2] Tou J. T, Gonzalez R. C, "Pattern recognition principle.", *Addison Wesley*, Reading, MA, 1974.

[3] Adetunmbi A.Olusola., Adeola S.Oladele. and Daramola O.Abosede, "Analysis of KDD'99 Intrusion Detection Dataset for Selection of Relevance Features.", *Proceedings of the World Congress on Engineering and Computer Science 2010 Vol I*, WECECS 2010, October 20-22, 2010.

[4] Dr.S.Vijayarani and Ms. Maria Sylviaa.S, INTRUSION DETECTION SYSTEM-A STUDY.",*International Journal of Security, Privacy and Trust Management (IJSPTM) Vol4*, No 1, February 2015

[5] K. Jayan and A. K. Rajan, "Sys-log classifier for complex event processing system in network security," in *Advances in Computing, Communications and Informatics (ICACCI, 2016 International Conference on IEEE)*, 2016, pp. 2031-2035.

[6] K. Jayan and A. K. Rajan,"Preprocessor for complex event processing system in network security," in *Advances in Computing, Communications (ICACC), 2014 Fourth International Conference on.* IEEE , 2014, pp. 187-189.

[7] P. M S, V. Hedge, H.Anushadevi, and V. Ambika,"Automated spam detection in email using svm," vol. 10, pp. 25219-25228, 01 2015.