การประยุกต์ใช้การจัดกลุ่มแบบสองชั้นเพื่อค้นหาผู้บุกรุกในล็อกขนาดใหญ่

นายจักรรินทร์ เทิดภาปิยะนาค

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

APPLYING DOUBLE CLUSTERING TECHNIQUE FOR INTRUSION DETECTION IN LARGE-SCALE LOG

Mr. Jakrarin Therdphapiyanak

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2012

Thesis Title	APPLYING DOUBLE CLUSTERING TECHNIQUE FOR INTRUSION DETECTION IN LARGE-SCALE LOG
By	Mr.Jakrarin Therdphapiyanak
Filed of Study	Computer Engineering
Thesis Advisor	Assistant Professor Krerk Piromsopa, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master’s Degree

…………………………………………….. Dean of the Faculty of Engineering

(Associate Professor Boonsom Lerdhirunwong, Dr.Ing.)

THESIS COMMITTEE

……………………………………………… Chairman

(Assistant Professor Natawut Nupairoj, Ph.D.)

…………………………………………….... Thesis Advisor

(Assistant Professor Krerk Piromsopa, Ph.D.)

……………………………………………… Examiner

(Assistant Professor Veera Muangsin, Ph.D.)

……………………………………………… External Examiner

(Pongtawat Chippimolchai, Ph.D.)

จักรรินทร์ เทิดภาปิยะนาค : การประยุกต์ใช้การจัดกลุ่มแบบสองชั้นเพื่อค้นหาผู้บุกรุกในล็อกขนาดใหญ่. (APPLYING DOUBLE CLUSTERING TECHNIQUE FOR INTRUSION DETECTION IN LARGE-SCALE LOG) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ.ดร.เกริก ภิรมย์โสภา, 81 หน้า.

ในงานวิจัยนี้ได้นำเสนอการประยุกต์ใช้การจัดกลุ่มแบบสองชั้นเพื่อค้นหาผู้บุกรุกในล็อกขนาดใหญ่ เพราะล็อกไฟล์ คือ ไฟล์ที่เก็บข้อมูลของการกระทำ, กิจกรรม และเหตุการณ์ต่างๆที่เกิดขึ้นในระบบ ในระบบคอมพิวเตอร์สมัยใหม่เป็นระบบที่มีขนาดใหญ่และมีความซับซ้อน ทำให้ล็อกไฟล์เหล่านี้มีปริมาณมหาศาลและมีขนาดใหญ่มาก ดังนั้นการนำข้อมูลเหล่านี้มาวิเคราะห์เพื่อหาความผิดปกติที่เกิดขึ้นกับระบบจึงเป็นวิธีการที่จะสามารถเพิ่มความมั่นคงปลอดภัยให้กับระบบได้มากยิ่งขึ้น และโดยทั่วไป รูปแบบของข้อมูลปกติจะมีอยู่เป็นส่วนมากของรูปแบบข้อมูลทั้งหมด ดังนั้น ในงานวิจัยนี้ได้ประยุกต์ใช้ขั้นตอนวิธีในการทำเหมืองข้อมูล (Data Mining) คือ K-Means Algorithm และ Parallel FP-Growth ด้วย Apache Mahout Framework เพื่อทำการจัดกลุ่มและค้นหารูปแบบของความสัมพันธ์ที่เกิดขึ้นบ่อยครั้งในล็อกไฟล์เหล่านี้ จากนั้นจึงสร้าง Normal Profiles ขึ้นมา เพื่อดึงรูปแบบของข้อมูลปกติออกจากรูปแบบข้อมูลทั้งหมด ดังนั้นข้อมูลส่วนที่เหลือจะเป็นข้อมูลที่มีความน่าจะเป็นที่จะเป็นผู้บุกรุก ข้อมูลเหล่านี้จะถูกนำมาจัดกลุ่มและค้นหาความสัมพันธ์อีกครั้งหนึ่งเพื่อให้ได้มาซึ่งลักษณะเฉพาะของการโจมตีเหล่านั้น ซึ่งลักษณะเฉพาะเหล่านี้เป็นองค์ความรู้ที่จะสามารถระบุถึงลักษณะของผู้บุกรุกที่มีอยู่ในล็อกขนาดใหญ่

ภาควิชา ....วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อนิสิต : ....

สาขาวิชา ......วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ....

ปีการศึกษา ...2555.............................

557 04764 21 : MAJOR COMPUTER ENGINEERING

KEYWORDS: LOG ANALYSIS / LARGE-SCALE LOG ANALYSIS / DISTRIBUTED LOG ANALYSIS / LOG ANALYSIS USING DOUBLE CLUSTERING / KDD'99

JAKRARIN THERDPHAPIYANAK : APPLYING DOUBLE CLUSTERING TECHNIQUE FOR INTRUSION DETECTION IN LARGE-SCALE LOG. ADVISOR : ASST.PROF.KRERK PIROMSOPA, Ph.D., 81 pp.

In this dissertation, we proposed an applying double clustering technique for intrusion detection in large-scale log. Log files are list of actions, events and activities that happened in the system. These data of log files are humungous and useless. Therefore, log analysis is another way to enhance the security of the system. K-Mean algorithm and Parallel FP-Growth based on Apache Mahout are applied to cluster these log files and discover the frequent patterns to generate the normal profiles respectively. After the normal patterns are generated, the normal records will be removed from the data set. Therefore, the remaining records are the suspect intrusion records. These remaining records are partitioned and analyzed once again. Finally, the characteristics of these suspect intrusion records are generated. These characteristics are new knowledge and useful to enhance the security of the system.

Department : Computer Engineering Student's Signature : ....

Field of Study : Computer Engineering Advisor's Signature : ....

Academic Year : 2012

Acknowledgements

This dissertation would not have been completed without the help of many people to whom I am forever indebted. First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Krerk Piromsopa, who provided invaluable guidance and assistance throughout my time as a student at Chulalongkorn University. Dr. Krerk was always available and willing to help with any problem.

I would also like to express my utmost gratitude to my thesis committee: Dr. Natawut Nupairoj, Dr. Veera Muangsin, and last but not least Dr. Pongtawat Chippimolchai. They provided essential guidance, especially in finishing my dissertation, and their invaluable comments, suggestions, and criticisms improved the quality of my dissertation immensely.

During my time at Chulalongkorn University, I have been very fortunate to enjoy the advice, support, and encouragement of my friends and colleagues. Thank you all, my friends.

Finally, I would like to thank my father and mother who always believed in me. I could not have done it without their support.

Contents

Page

List of Tables

Page

Page

List of Figures

CHAPTER I
INTRODUCTION

Log files are data which contain system's activities. They also contain user's activities. These data are tremendous and useless. Therefore, they are used for investigating the cause of a problem. Another alternative to make these log files more useful is to extract knowledge from them. Hence, an analysis is required to extract knowledge from such data.

Given the popularity of internet, security has become one of the most important issues in every system. Log files are not only used for tracing anymore but they are used to enhancing system security. Log analysis is another way to enhance system security. With the analyzing and determining these log files, the new knowledge that cannot explicitly be seen with simple analysis is obtained.

In a small system, the incoming traffic logs are coped with the standalone IDS or IPS. In a large system a large amount of traffic logs tends to exceed the capacity of a single IDS node. Therefore, a distributed log analysis system has implemented to support high volume of log files.

This chapter is divided into 7 parts: problem statement, objectives, scopes of study, definitions, expected results, published proceedings and organization of the dissertation.

1.1 Problem Statement

Log Files are data which contain list of system's events and system's activities. If there is any event occurs in the system, log files are generated. In large-scale systems, such as distributed systems, clusters and grid systems, huge amount of log data are generated. Their tremendous size is one of the vulnerabilities of the system because of the standalone log analyzer cannot deal with their humungous size. The standalone cannot analyze them all. Even though standalone log analyzers can analyze all log files, the results might have errors.

With the tremendous size of log files, log files are only used for trace analysis. However, trace back is an investigation not prevention method. Hence, using log files for analysis is more beneficial than using log files for trace back.

From the reasons above, log analysis is another way to enhance system security. The performance of a distributed log analysis system must adapt to match the size of log files.

1.2 Objectives

In our study, we aim to create the system that has the following features.

1. Support high volume of data traffic and use less time than standalone log analysis methods with the same size of log files
2. Have the ability to identify probabilities of results of the analysis that show opportunity of them to be an intruder
3. Able to screen log files to create new knowledge, which has a high probability to be an intruder, that we can use to create firewall rules
4. Have the ability to distribute processes and analyze log files using Hadoop

1.3 Scopes of study

1. This project only uses the anomaly detection method.
2. This project does not focus on a real-time log analysis.
3. The results of the analysis show only the deviated behaviors which have high probability of being an intrusion.
4. The result of this project is the new knowledge that can be used to develop firewall rules.
5. The purpose of this project is to find new knowledge from large amount of log files, not to create preventive measures.

1.4 Definitions

1. Intrusion

Intrusion is a group of actions or activities that attempts to compromise the confidentiality, integrity, or availability (CIA) of the system [1].

2. Intrusion Detection System (IDS)

Intrusion detection system (IDS) is a system that detects intrusions, intrusion attempts and malicious activities in real time on network traffics and makes an active decision on those intrusions as a preventive measure to protect the system against those threats [2].

3. Misuse Detection

Misuse Detection is a detection method based on known malicious patterns in database [1].

4. Anomaly Detection

Anomaly detection is a detection method based on unknown pattern. These intrusions are set of activities that their behaviors are deviate from the normal behavior. However, these intrusions are only possible intruders [1].

5. Apache Hadoop

Apache Hadoop [3] is a framework for the distributed system and cluster for processing large set of data using a MapReduce programming model. Its concept is to process large data sets with thousands of machines in a cluster instead of a single server. Hadoop provides high availability, fault tolerance and scalability. Hadoop has three subprojects: Hadoop Common, Hadoop Distributed File System (HDFS) and Hadoop MapReduce.

6. Apache Mahout

Apache Mahout [4], [5] is one of Apache projects that provides scalable machine learning algorithms such as clustering, classification, association rule mining and batch based collaborative filtering etc. These algorithms are implemented on a top of Hadoop using MapReduce Programming model.

7. Clustering

Clustering is one of data mining techniques for partitioning data and assigning them into groups with same characteristics [1].

8. Hadoop MapReduce

MapReduce is a programming paradigm for distributed process large data sets proposed by Google [6].

9. Association Rule Mining

The association rule mining is one of data mining techniques used to find relationship among items in large data sets that frequently appear together [1].

1.5 Expected Results

In our study, we expect the results hereinafter

1. Can identify high probability intruder from high volume of log files.
2. Can apply data mining technique to enhance the security of the system.
3. Can apply Hadoop with log analysis to enhance performance issue.
4. Can apply association rule technique to emphasize the possible intruder.
5. Provide new knowledge from large amount of log files toward firewall rules.

1.6 Published Proceedings

A part of this dissertation is published in the ICUIMC 2013 Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication in title of “Applying Hadoop for log analysis toward distributed IDS” [7] proposed by Jakrarin Therdphapiyanak and Krerk Piromsopa. This conference is organized at Kota Kinabalu, Malaysia on 17-19 January 2013.

Another part of this dissertation is published in the ECTI-CON 2013 Proceedings of 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology in the title of “An analysis of suitable parameters for efficiently applying K-Means clustering to large TCPdump data set using Hadoop framework” [8] proposed by Jakrarin Therdphapiyanak and Krerk Piromsopa. This conference is organized at Krabi, Thailand on 15-17 May 2013.

1.7 Organization of the Dissertation

This dissertation is organized as follows: Chapter 2 is the related works. Chapter 3 discusses a system design. Chapter 4 describes our implementation. Chapter 5 shows our experimental results and the last Chapter is the conclusion of our works.

CHAPTER II
RELATED WORK

In this chapter, we describe related theories in the Section 2.1 and related research in the Section 2.2.

2.1 Related Theories

This Section is divided into 7 parts: intrusion detection system, K-Means clustering algorithm, FP-Growth algorithm, Confusion Matrix, Receiver Operating Characteristic (ROC) Curve, Silhouette Index and Apache Hadoop together with its subprojects.

2.1.1 Intrusion Detection System (IDS)

Intrusion Detection System (IDS) [9], [10] is a system that inspects and examines network traffics for malicious activities. The primary objective of an Intrusion Detection System is to alert the system administrator for any abnormal activity that occurred and to notify the system administrator to deploy preventive measures against those threats.

We can classify Intrusion Detection System into two groups by the detection method that they used:

1. Misuse Detection

This detection method detects intrusions by using known malicious patterns in a database. The method is pretty accurate at detecting known threats, but cannot detect any unknown malicious patterns.

2. Anomaly Detection

This detection method is well suited for detecting unknown attacks. The method can detect and examine patterns of activities that deviate from the norm. The result of this detection method usually shows possible intruders.

Another way of classifying Intrusion Detection System is by their working behavior.

1. **Host-based IDS**

   Host-based IDS detects and examines traffics on each host in the system for any malicious activities by analyzing packets on each host network interfaces.

2. **Network-based IDS**

   Network-based IDS detects and examines network traffic of the system by analyzing network protocols for any suspicious activities.

2.1.2 K-Means Clustering Algorithm

K-Means algorithm [1], [11], [12], [13] is one of the most commonly used clustering algorithms. The primary usage of a clustering algorithm such as K-Means is to partition data into groups where each group contains data which have the same characteristics. K-Means is described by the following equation:

$$E = \sum_{i=1}^{k} \sum_{x \in C_i} distance^2(x, m_i)$$

Where *x* is an observation data set (x_1, x_2, x_3, … x_n) in cluster C_i and m_i is the centroid (mean) of C_i cluster. K-means partitions these n observations into predefined *k* clusters ($k \leqslant n$). The distance (*E*) is calculated from each point to the centroid of each cluster. Each observation will be assigned to the nearest cluster centroid measured by distance. The set of cluster (C_1, C_2, C_3, …,C_k) is the result after partitioning.

After distances are calculated and each data point is assigned into cluster, the centroid of each cluster is re-computed. The process is repeated until the centroid of each cluster stops changing.

2.1.3 FP-Growth Algorithm

Frequent Pattern Growth (FP-Growth) algorithm [13], [14], [15] is an association rule mining algorithm. FP-Growth algorithm is typically used to find frequent patterns from the FP-Tree data structure. The advantage of FP-Growth over Apriori [13], [14] lies in the fact that FP-Growth allows the discover of frequent item sets without generating candidate item sets, which unlike Apriori which has to generate candidate item sets and then find the frequent item sets later. FP-Growth algorithm consists of the following 2 steps:

1. **Construction of FP-Tree**

   In this step, the database is scanned 2 times. For the first scan, the frequent items are selected to form F-list. Then, the FP-Tree is constructed in the second scan

2. **Mining Frequent Patterns from FP-Tree**

   For this step, every node in FP-Tree is traversed by the FP-Growth algorithm. During FP-Growth visit each node, it obtains the prefix path sub tree of each node. The prefix path sub tree of each node calls Conditional Pattern Base of each node. Conditional Pattern Base is sub tree of the interesting node which occurs together with that node. After that, the FP-Tree is constructed from the Conditional Pattern Base and the frequent patterns are mined again. This process recursively occurs until cannot generate more Conditional Pattern Base.

2.1.4 Confusion Matrix

Confusion matrix [16] which is used to evaluate and validate the classifier system is a specific table containing 4 values. They are (1) true positives, (2) true negatives, (3) false positives and (4) false negatives. The performance of the algorithm can be represented by these values. The example of the confusion matrix is shown in Figure 2.1. Each value can be described by comparing the predicted labels to the real labels as follows:

Positive Conditions	Negative Conditions	
True Positive (The attacks which are correctly identified as intrusion) ***(TP = 30)***	False Positive *(The normal patterns which are incorrectly identified as the intrusion)* *(FP = 10)*	Positive Predictive Value (Detection Rate) = TP / (TP + FP) = 30 / (30 + 10) = 75%
False Negative *(The normal patterns which are incorrectly identified as the intrusion)* ***(FP = 10)***	True Negative *(The normal patterns which are correctly identified as normal activities)* ***(TN = 50)***	Negative Predictive Value (NPV) = TN / (FN + TN) = 50 / (10 + 50) = 83.33%
Sensitivity (TPR) = TP / (TP + FN) = 30 / (30 + 10) = 75%	Specificity (TNR) = TN / (FP + TN) = 50 / (10 + 50) = 83.33%	

Figure 2.1: The example of the confusion matrix

1. **True Positives (TP)** is the percentage of attack patterns which are correctly identified as the intrusion.

2. **True Negatives (TN)** is the percentage of normal patterns which are correctly identified as the normal activities.

3. **False Positives (FP)** is the percentage of normal patterns which are incorrectly identified as the intrusion.

4. **False Negatives (FN)** is the percentage of attack patterns which are incorrectly identified as the normal activities.

With these 4 values, TP, TN, FP and FN, from the confusion matrix, the other statistical measure values can be derived such as true positive rate (TPR) or sensitivity, false positive rate, true negative rate (TNR) or specificity, false negative rate, Positive predictive value (PPV) or Detection rate and accuracy. These values are represented as follows [17]:

1. **True positive rate (TPR) or sensitivity** is the proportion of attack patterns which are correctly identified as the intrusion to the all exactly attack patterns.

$$TPR\ (Sensitivity) = \frac{TP}{(TP + FN)}$$

2. **False positive rate (FPR)** is the proportion of normal patterns which are incorrectly identified as the intrusion to the all exactly normal patterns.

$$FPR = \frac{FP}{(FP + TN)}$$

3. **True negative rate (TNR) or specificity** is the proportion of normal patterns which are correctly identified as the normal activities to the all exactly normal patterns.

$$TNR\ (Specificity) = \frac{TN}{(TN + FP)}$$

4. **False negative rate (FNR)** is the proportion of attack patterns which are incorrectly identified as the normal activities to the all exactly attack patterns.

$$FNR = \frac{FN}{(FN + TP)}$$

5. **Accuracy** is the proportion of all the correctly identified patterns to all the correctly and incorrectly identified patterns.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

6. **Positive predictive value (Detection rate)** is the proportion of attack patterns which are correctly identified as the intrusion to all the predicted attack patterns.

$$PPV = \frac{TP}{(TP + FP)}$$

The performance of the system is usually presented in the terms of detection rate and false positive rate (false alarm rate). However, the Receiver Operating

Characteristic (ROC) or ROC curve [18], [19] is also used to represent the performance of a binary classifier system by the fraction of true positive rate (TPR) and false positive rate (FPR).

2.1.5 Receiver Operating Characteristic (ROC) Curve

A Receiver Operating Characteristics (ROC) curve [18], [19] is a graph which presents the performance of the classifier system. ROC graph is plotted by the fraction of true positive rate (TPR) and false positive rate (FPR). ROC curve is useful to determine the appropriate threshold for the system.

ROC curve is commonly used in the classifier system such as the medical study and machine learning. However, we apply ROC curve with our system which is applied clustering algorithm. ROC curve is applied to determine the performance model of the Double Clustering techniques. The Double Clustering technique is our proposed method for the intrusion detection which use K-means algorithm. By applying ROC to K-Means, ROC graph is plotted by the fraction of the various number of K and the result of each K.

2.1.6 Silhouette Index

Silhouette index [20], [21], [22] is one of validation methods for clustering techniques. The quality of the clustering can be evaluated by the Silhouette index. Silhouette index indicates that the clusters are properly partitioned or not. Silhouette index (S_i) equation is showed in expression (1) [20], [21], [22]:

$$\boldsymbol{S_i^j = \frac{b_i^j - a_i^j}{\max(a_i^j, b_i^j)}} \tag{1}$$

From the expression (1), the values of S_i^j are ranges between -1 and 1. For value 1 of S_i^j, it indicates that the clusters are properly grouped. The data points are closer to points in their own cluster than points in neighbor clusters. In contrast, if S_i^j value is equal to -1, it indicates that data points are closer to other neighbor clusters than their own clusters. Therefore, Silhouette index is used to compare the result of the clustering that the data points are properly partitioned or not.

As shown in the expression (1), Silhouette index consists of two variables for calculating. There are a_i and b_i variables. a_i variable is the average distance from i-th data point to other data points in the same cluster. b_i variable is the average distance from the i-th data point to all data point in the nearest cluster. a_i and b_i is given by the following expression [20], [21], [22]:

$$\boldsymbol{a_i^j = \left(\frac{1}{m_j-1}\right)\sum_{\substack{k=1\\k\neq i}}^{m_j} d(x_j^i, x_j^k)\,, \quad 1,2,3,\dots,m_j} \tag{2}$$

$$\boldsymbol{b_i^j = \ min_{\substack{n=1,\dots,K\\n\neq j}}\left(\frac{1}{m_n}\sum_{k=1}^{m_n} d(x_j^i, x_k^n)\right), \quad 1,2,3,\dots,m_j} \tag{3}$$

Assuming that x is an observation data set $(x_1, x_2, x_3, \dots, x_n)$ and C is a set of cluster $(C_1, C_2, C_3, \dots, C_n)$. Let $d(x_i, x_j)$ be the distance between two data points x_i and x_j. Assuming that we have K clusters each cluster represents by C_j where $j = (1, 2, 3, \dots, K)$. The number of members in each cluster represents by m_j where m_j is a number of members in the j-th cluster. The members of the j-th cluster is represents by $(x_1^j, x_2^j, x_3^j, \dots, x_{m_j}^j)$.

From the expression (2), a_i^j is the average distance from the i-th data points in the j-th cluster to others data points in its own cluster. From the expression (3), b_i^j is the average distance from the i-th data point in the j-th cluster to all data points in the cluster which is the nearest cluster to the j-th cluster.

With the a_i^j and b_i^j of the j-th cluster in hand, the Silhouette index of the i-th data point in th j-th cluster is calculated by the expression (1). Therefore, Silhouette index of every data point in the j-th cluster is calculated. After that, the Silhouette index of the j-th cluster is calculated by the expression (4) [20], [21], [22].

$$\boldsymbol{S_j = \frac{1}{m_j}\sum_{i=1}^{m_j} S_i^j} \tag{4}$$

From the expression (4), Silhouette index of the j-th cluster is calculated. With the Silhouette index of every cluster in hand, the global Silhouette index is presented to evaluate the results of the clustering. The global Silhouette index is shown in the expression (5) [20], [21], [22]. It indicates the quality and the properly of the clustering results.

$$S = \frac{1}{K} \sum_{i=1}^{K} S_j \tag{5}$$

There is another index which is used to evaluate and validate the quality of the result of the clustering. It is Davies-Bouldin index [20], [23], According to the experiments in [20], their experiments show that Silhouette index produces more accurate results than that of the Davie-Bouldin index. With the better result of accuracy, Silhouette index uses much more computation time than that of Davies-Bouldin index. Moreover, the Silhouette index's computation method is more complex than the Davies-Bouldin method.

2.1.7 Apache Hadoop and its Subprojects

Apache Hadoop [3] is a framework for the distributed system for processing large data based on MapReduce programming model. MapReduce [6] is a programming paradigm for distributed process with large data sets. The main concept of the Apache Hadoop is to process large data sets with thousands of machines in a cluster instead a single server. The prominent of Hadoop are that it provides high availability, fault tolerance and scalability. Moreover, Hadoop has three subprojects. They are (1) Hadoop Common, (2) Hadoop Distributed File System (HDFS) and (3) Hadoop MapReduce.

Another project of Apache Hadoop is Apache Mahout [4], [5] which is an Apache library that provides scalable machine learning algorithms such as classification algorithm, clustering algorithm and Association rule mining. These algorithms are implemented on a top of Hadoop using java based on MapReduce paradigm.

2.2 Related Research

There are many researches in the field of intrusion detection system. These researches can be grouped into 5 groups as follows:

2.2.1 A survey on data mining techniques for Intrusion Detection

In [1], a group of researchers presented the survey on intrusion detection systems using data mining techniques for effective detection of misuse detection and anomaly detection. Working of intrusion detection system can divide into 4 steps.

1. **Data Collection**: collects network traffic.
2. **Feature Selection**: select only necessary attribute.
3. **Analysis**: determine whether data is malicious or not.
4. **Action**: alarm the system administrator to make decision on the normal and abnormal activities and provide preventive measures.

Moreover, this paper had proposed that classification methods commonly used for misuse detection and clustering methods can be applied to both anomaly detection and misuse detection. As compare with these two methods, classification technique is less efficient than clustering technique in the field of intrusion detection. The last data mining technique, association rule mining, is very useful in intrusion detection

2.2.2 Clustering Techniques for Intrusion Detection

In [24], [25], [26], they apply clustering algorithm for the intrusion detection. KDD'99 data set is used as the evaluation data set. Their Experiments showed that clustering algorithms gave the good results in the field of anomaly detection.

In [24], the mixed intrusion detection system was proposed. The mixed intrusion detection system consists of misuse and anomaly detection. For the anomaly detection module, K-means algorithm was applied. They generated four data sets which each data set has two thousand and one hundred records for their experiments. There are two thousand records of normal data and one hundred records of intrusion data in each data set. The results of their experiments with K-means algorithm produced high detection rate for a single intrusion but produce low detection rate for multiple intrusions. Therefore, the improved K-means algorithm namely KD algorithm is proposed. They proposed KD algorithm and evaluate it with the same data set. The results of the experiments with the KD algorithm produced high detection rate for either single intrusion or multiple intrusions. However, the data set for their experiment has only two thousand and one hundred entries.

The distributed intrusion detection system (IDS) was proposed in [25], a group of researchers applies K-means algorithm based on distributed model for clustering. They evaluated their system with the KDD'99 data set. They resampled data sets for their experiments. The resampled data sets consisted of 98.5 to 99% of normal data and the rest is intrusion data. The Receiver Operation Characteristic (ROC) curve was used to evaluate their experiments.

Another research in [26] proposed the intrusion detection model using Fuzzy C-means algorithm. KDD'99 data set is used for the evaluation. Their experiments showed that Fuzzy C-means algorithm is effective for anomaly detection. Moreover, their experiments showed that the false alarm rate and detection rate increase when the number of clustering centers increased.

2.2.3 Classification Techniques for Intrusion Detection

Classification techniques are commonly applied for the intrusion detection system. With their learning phase, classification techniques yield the good results for intrusion detection. However, these techniques are appropriate with the anomaly detection method.

In [27], three classification algorithms are compared for the intrusion detection. There are (1) C5.0 Decision Tree, (2) Ripper Rule and (3) Support Vector Machine (SVM). They researched and compared to determine the most efficient algorithm among these three algorithms. Moreover, KDD'99 data set is used for their evaluation. The results of their experiments showed that C5.0 Decision Tree produced the most accurate results with the detection rate higher than 96%.

Another research in [28] proposed a new approach to improve the detection rate of the intrusion detection system. A group of researchers apply fuzzy neural network and Support Vector Machine (SVM) algorithm for the intrusion detection. However, K-Means clustering is applied at the first step of their approach. They evaluated their system with KDD’99 data set. The results of their experiments indicated that their approach produced the accuracy rate greater than 97%. Moreover, their approach can detect all attack types (DoS, PROBE, U2R, R2L) in KDD'99 data set.

2.2.4 Association Rule Mining to Enhance Intrusion Detection

Association rule mining is another technique to enhance intrusion detection. Apriori algorithm and FP-Growth algorithm are one of the most popular association rule mining. They are common used in the field of intrusion detection. They can reveal the interesting relations among a large set of data. So, in many researches, they are used to retrieve the interesting relations in a large data set.

In [2], they proposed a method to generate real-time firewall rules by using Snort and Apriori algorithm. Snort is used to record log files of user's activities. Snort itself is network-based Intrusion detection software using rule sets to detect intrusions. After Snort intrusion detection processes, association rule, an Apriori algorithm, is applied to create a model to detect activities (IP or port) which are malicious by calculate their support and confidence numbers. If their support and confidence number exceeds a defined threshold, firewall rules are generated.

2.2.5 MapReduce Model for Log Analysis and Intrusion Detection

In [29], a group of researchers proposed system anomaly detection using MapReduce programming model to analyze the distributed log. Log files in this paper are system logs from each node in a cluster such as CPU, IO, network, memory, etc. Mahout K-Means clustering algorithm is used for clustering logs from the same node. Then, they get records of same status types using MapReduce programming model. Finally, they show visual report and visualized graph as a result.

In this paper, they compared their contribution with existed system monitored tools such as vmstat, iostat, netstat, etc. After their experiment, they found that it is efficient to apply MapReduce model to analyze high volume logs.

In [30], they improved Apriori algorithm by using MapReduce model. Traditional Apriori algorithm consists of two steps:

1. Generating all frequent item sets.
2. Calculating confident of each frequent item set.

The technique they proposed is to use MapReduce model to generate all frequent item sets simultaneously. Their experiments showed the speedup of their parallel Apriori based on MapReduce with large data sets. In contrast to the small data sets, performance of their parallel Apriori cannot overcome an overhead (including communication time) of MapReduce model.

In [14], FP-Growth is one of the association mining rules like Apriori algorithm. FP-Growth based on MapReduce model proposed by Apache Mahout as a parallel frequent pattern mining, parallel FP-Growth algorithm. FP-Growth can overcome Apriori algorithm because Apriori needs to generate all frequent item sets and scan data sets many time differ from FP-Growth. FP-Growth consists of two steps:

1. Construction of FP-Tree.
2. Generation of frequent item sets from FP-Tree.

This paper proposed an improvement of the parallel FP-Growth algorithm by using balanced strategy and generates all frequent item sets unlike parallel FP-Growth algorithm. The results of their experiment showed that parallel FP-Growth algorithm with balanced strategy has an approximate 1.5 speedup compare to parallel FP-growth algorithm.

CHAPTER III

SYSTEM DESIGN

This research proposes a distributed log analysis system using Hadoop to provide new knowledge that can be used to develop firewall rules using data mining techniques. We use K-Means clustering algorithm based on MapReduce programming for anomaly detection and use parallel FP-growth algorithm to find relations between data in a large data set of possible intruder clusters. Our design can be described in Figure 3.1.

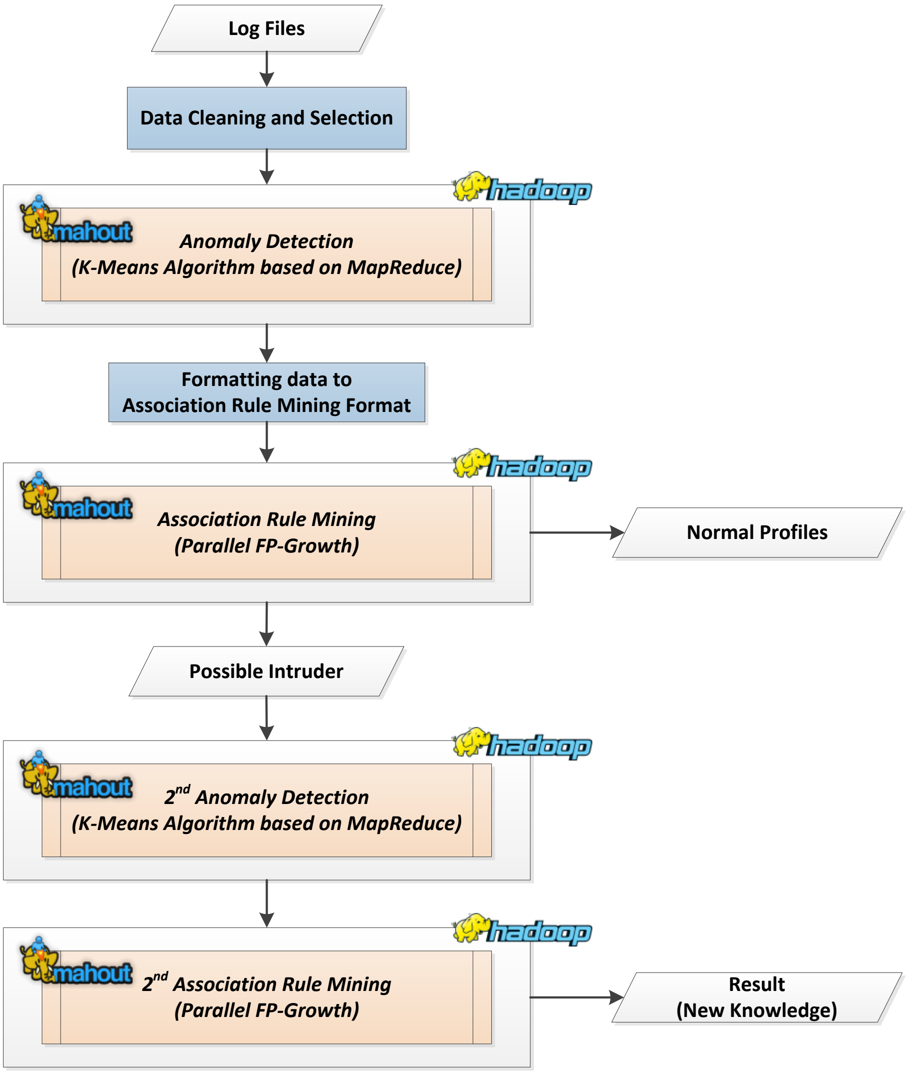


Figure 3.1: Working Flow of Distributed Log Analysis using Mahout

From Figure 3.1, our system is divided into 2 main procedures. The first procedure is to construct the normal profiles. So, the predicted normal records will be removed from the system. The rest of the records labeled as possible intruders are sent to the next procedure. The second procedure is to retrieve the characteristics of those possible intruder records. Each component in our system can be described as follows:

3.1 Log Files

The input of the system is log files. KDD'99 data set [31] is used as an evaluation data set. KDD'99 Training data set is used to evaluate our design. Then, KDD'99 Test set and common log files from an Apache server are used to evaluate our system.

3.2 Data Cleansing and Selection

Before log files are sent to analysis phase, which is the phase of anomaly detection using K-Means algorithm based on MapReduce programming, data cleansing and selection process is necessary. First, the duplicated entries are removed. Then, each attribute and each entry have to be converted to number. After the data cleansing and selection, the log files are ready to be analyzed.

3.3 Anomaly Detection

In this step, K-Means algorithm based on MapReduce using Apache Mahout is used to partition log entries into a group with the same properties of log files. After partitioning, we assume that the same properties of records will be grouped together. Data will be grouped into K clusters. With these K cluster, the probability to generate the various normal behaviors will increase.

3.4 Formatting Data to Association Rule Mining Format

The result of anomaly detection will be sent to association rule mining step. Thus, data have to be cleaned and selected again. This process will transform data into an appropriate format for the parallel FP-Growth algorithm.

3.5 Association Rule Mining

All records are grouped into K clusters. Therefore, we will find the frequently occurring relations of each cluster using parallel FP-Growth algorithm. In this step, the normal behaviors from each cluster are generated. The defined threshold in this step is 70% [2]. If the probability of occurrence is more than the defined threshold, that relation item set is labeled as the normal profiles of the system.

3.6 Normal Profiles

The normal profiles of the system are the frequently occurring patterns. So, these normal profiles are used as signatures of the system. So, the entire input log files will be compared to the signatures. If the record matches to any signature, that record will be removed and labeled as the normal record.

3.7 Possible Intruder Records

After all of the records are compared to the signatures, the normal records will be removed. So, the rest of the records are labeled as a possible intruder records. These records will be analyzed in the next process to retrieve their characteristics.

3.8 The 2nd Anomaly Detection

In this step, K-Means algorithm based on MapReduce using Apache Mahout is applied again. The possible intruder records are the input of this step. All records will be partitioned into K clusters. After the partitioning, the largest cluster is labeled as an intrusion. If any remaining clusters are larger than one-fourth of the largest cluster, that cluster will be labeled as an intrusion. The cluster that is smaller than one-fourth of the largest cluster will be labeled as outliers and not included for analyze in the next process.

3.9 The 2nd Association Rule Mining

With the possible intrusion cluster in hand, parallel FP-Growth algorithm is used to retrieve their frequently occurring relations. The defined threshold is 70% [2]. After this process, the characteristics of the possible intrusion are generated.

3.10 Result

The results of the system are a new knowledge of anomaly detection, which has a high probability to be an intruder and can be used to interpret and be applied to create firewall rules.

CHAPTER IV
IMPLEMENTATION

This chapter describes the implementation of the “Applying Double Clustering Technique for Intrusion Detection in Large-Scale Log”. This chapter is divided into 7 Sections. Section 4.1 describes the details of the KDD’99 data set which we use for the evaluation. Before the analysis phase, the initial preprocessing process is shown in Section 4.2. Then, Section 4.3 discusses the results of K-Means clustering. Section 4.4 shows the results of Parallel FP-Growth algorithm. Section 4.5 and Section 4.6 describe the results of the 2nd K-Means clustering and the 2nd Parallel FP-Growth algorithm respectively. The last Section is the conclusion of the implementation.

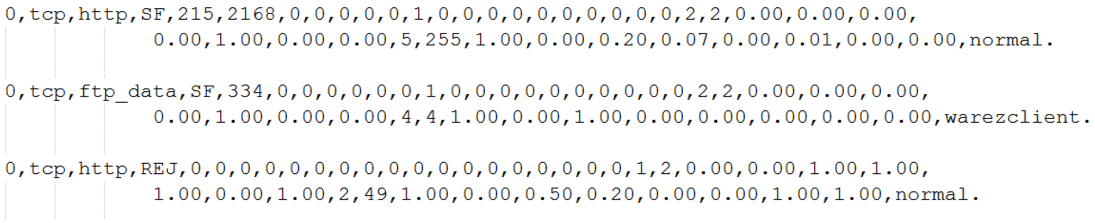
4.1 KDD’99 Data Set

KDD'99 data set [31] is used as input data for the implementation and experiments. KDD’99 data set is generated by MIT Lincoln Labs for nine weeks by simulating a typical U.S. Air Force local-area-network (LAN) with a true Air Force environment. Moreover, KDD'99 data set is a raw TCP dump data which has approximately five million entries for the full data set and has approximately five hundred thousand entries for the ten-percentage subset. Each record of data set consists of 41 attributes.

There are four major attack types in KDD’99 data set as follows:

1) **Denial of Service (DoS)** such as SYN flood
2) **Remote to Local (R2L)** is an unauthorized access from a remote machine such as guessing password
3) **User to Root** is unauthorized access to local super user (root) privileges such as buffer overflow attacks
4) **Probing** such as port scanning

One of the major characteristic of KDD’99 data set is the fact that each record is labeled as either a normal or a specific attack type. There is one label for each record these labels are classified into one of the four categories. The example of KDD’99 data set is shown in the Figure 4.1.



```
0,tcp,http,SF,215,2168,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,
            0.00,1.00,0.00,0.00,5,255,1.00,0.00,0.20,0.07,0.00,0.01,0.00,0.00,normal.

0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,
            0.00,1.00,0.00,0.00,4,4,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,warezclient.

0,tcp,http,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,1.00,1.00,
            1.00,0.00,1.00,2,49,1.00,0.00,0.50,0.20,0.00,0.00,1.00,1.00,normal.
```

Figure 4.1: The example of KDD’99 data set

The example of the KDD’99 data set is shown in Figure 4.1. Each record has 41 attributes and the last attribute is the label of that record. The last attribute indicates that the record is a normal record or the attack record with the one specific type for each record. The last attribute which is the label attribute can be used for evaluation and validation.

The KDD’99 data set is divided into two set of data. There are (1) the training set of KDD’99 and (2) the testing set of KDD’99. Each set of data is separated into two groups. They are the ten-percentage subset of KDD’99 and the full data of KDD’99 data set. Therefore, there are 4 parts of the KDD’99 data set. The training set of KDD’99 is used for the implementation process. For the experiments process, the testing set of KDD’99 is used.

In the implementation process, the appropriate parameters such as the appropriate number of K, the proper number of iterations, the efficient length of rule are determined. In the experiment process, the testing set of KDD’99 is used to evaluate our system. Finally, the results of our experiments are shown and compared to other researches.

4.2 Initial Preprocess

The initial preprocess is divided into 2 parts: reducing the number of entries and encoding attributes as numbers.

4.2.1. Reducing the number of entries

In this step, the duplicated records are removed. As shown in the Table 4.1 [32], it shows the original records of the full data set of KDD'99 which has approximately five million records. After the redundant records are removed, the number of record is reduced. Finally, there are approximately one million and seventy five thousand remaining records.

Table 4.1: The redundant records and ratio of the distinct records of the full data of KDD'99 training data set [32]

Labeled	Original Records	Distinct Records	Reduction Rate	Ratio of Distinct Records
Normal	972,781	812,814	16.44%	75.61%
Attacks	3,925,650	262,178	93.32%	24.39%
Total	4,898,431	1,074,992	78.05%	100.00%

There is another data set of the KDD'99 training set, namely the ten-percentage subset. There are approximately one hundred and fifty thousand records in the ten-percentage subset of KDD'99 training set. However, the ten-percentage subset of KDD'99 training set is also reduced the redundant records. The details of the ten-percentage subset and the reduction rate of the duplicated records are shown in the Table 4.2.

Table 4.2: The redundant records and ratio of the distinct records of the ten-percentage subset of KDD'99 training data set

Labeled	Original Records	Distinct Records	Reduction Rate	Ratio of Distinct Records
Normal	97,278	87,832	9.71%	60.33%
Attacks	396,743	57,754	85.44%	39.67%
Total	494,021	145,586	70.53%	100.00%

As shown in the Table 4.1 and Table 4.2, the reduction rate of the distinct records from the original records in the full data of KDD'99 training set and the ten-percentage subset of KDD'99 training set is approximately 78.05% and 70.53%

respectively. The ratio of the normal records from all of the records in the full data of KDD’99 training set and the ten-percentage subset of KDD’99 training set is approximately 75.61% and 60.33% respectively. So, the distinct records of the full data of KDD’99 training set and the ten-percentage subset of KDD’99 training set are both used for the implementation.

4.2.2. Encoding attributes as numbers

In this step, each attribute and each record have to be converted to number. The number is an appropriate format for clustering. Moreover, each attribute is multiplied with the constant number. After that, the distance from each record to the centroid will be calculated using Euclidean distance. The encoding and multiplying process is the process to increase the distance from each record to the centroid and makes the clustering easier.

4.3 Clustering using K-Means algorithm

In this process, K-means algorithm based on MapReduce paradigm is applied to partition data into K groups. After the clustering, the results of the clustering are evaluated and validated using the confusion matrix. However, K-means algorithm needs some parameters with its processing. Therefore, the appropriate number of initial cluster (K) and the appropriate number of iteration which are the essential parameters for K-means algorithm will be discussed.

4.3.1. The Appropriate Number of K

In this part, the appropriate number of K is discussed. The distinct full data of KDD’99 training data set and the ten-percentage subset of KDD’99 training data set are used for the experiments. Moreover, five data sets are generated by resampling from the ten-percentage subset of KDD’99 training set. Each sampling set from the ten-percentage subset of KDD’99 has twenty thousand records. In addition, five data sets are generated by resampling from the full data set of KDD’99. Each sampling set from the full data set of KDD’99 has two hundred thousand entries. The conclusion of the number of records of all the testing data sets is shown in Table 4.3.

Table 4.3: The details of the number of records of each testing data set

Data Set	Number of Records			
	Normal	Attacks	Total	percentage of Normal
10 percentage	87,832	57,754	145,586	60.33%
10per.Samp.01	12,112	7,888	20,000	60.56%
10per.Samp.02	12,038	7,962	20,000	60.19%
10per.Samp.03	12,059	7,941	20,000	60.30%
10per.Samp.04	11,714	8,286	20,000	58.57%
10per.Samp.05	8,310	11,690	20,000	41.55%
Full data set	812,814	262,178	1,074,992	75.61%
Full.Samp.01	83,295	116,705	200,000	41.65%
Full.Samp.02	150,751	49,249	200,000	75.38%
Full.Samp.03	102,980	97,020	200,000	51.49%
Full.Samp.04	151,184	48,816	200,000	75.59%
Full.Samp.05	72,075	127,925	200,000	36.04%

The details of the ten-percentage subset and full data of KDD'99 are shown in the Table 4.3. The K-means algorithm based on Apache Mahout is applied to cluster them. For the clustering, we run K-Means with number of K ranging from 2 to 50. The maximum iteration is 300 iterations for each number of K.

4.3.1.1. The Ten-Percentage Subset of KDD’99

The results of the clustering for the ten-percentage subset of KDD’99 and its sampling set are shown in the Figure 4.2. It shows the accuracy rate with the various numbers of initial clusters (K).

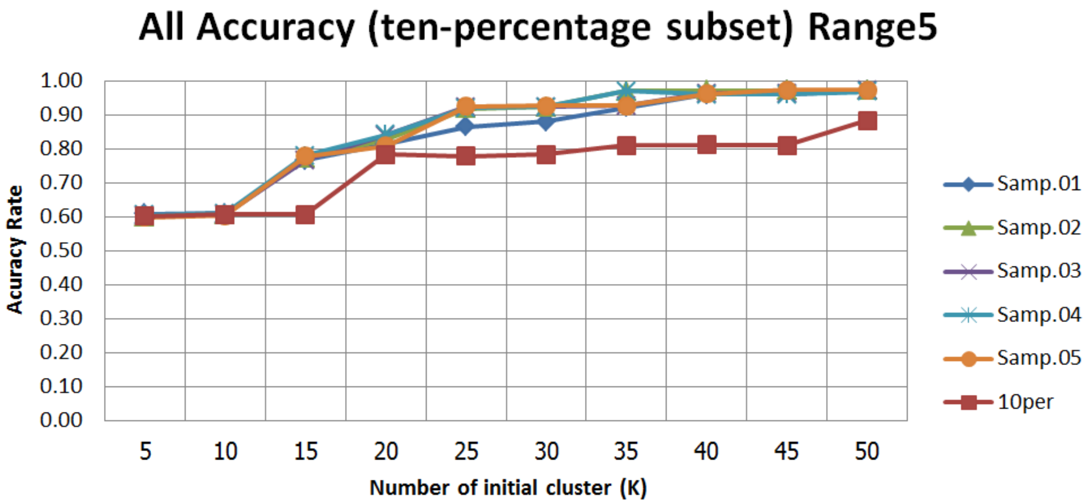


Figure 4.2: The accuracy graph of the ten-percentage subset of KDD’99 distinct training data set and its sampling data with number of K ranging from 2 to 50 (period of 5)

According to the Figure 4.2, all the test set has approximately 0.60 of accuracy rate at the beginning. The accuracy rate increases when increasing the number of K. The accuracy rate is close to 0.80 since K20. However, K25 is determined to be an appropriate number of K. There are four sampling data sets which has accuracy rate of 0.92. Another sampling data set has approximately 0.86 of accuracy rate. In contrast, a ten-percentage subset of KDD'99 data sets has approximately 0.78 of the accuracy rate.

According to the graph, the appropriate number of initial clusters (K) is 25. However, the accuracy rate decreases if the number of records increases. For the detailed information, ROC curve of the ten-percentage subset of KDD’99 and its sampling data sets are presented.

The accuracy rate is used to evaluate with the labeled data. For the other data sets which do not have the labeled for each record, the Silhouette Index is applied to evaluate the appropriate number of K instead of the accuracy rate. The result of the silhouette index is between -1 and 1. For the value which is nearby 1, it indicates that the data are properly clustered. In contrast, if the value is nearby -1, it indicates that the data are not properly clustered. Therefore, the results of the Silhouette index and the various numbers of initial K cluster of the sampling set of the ten-percentage subset of KDD’99 is shown in Figure 4.3.

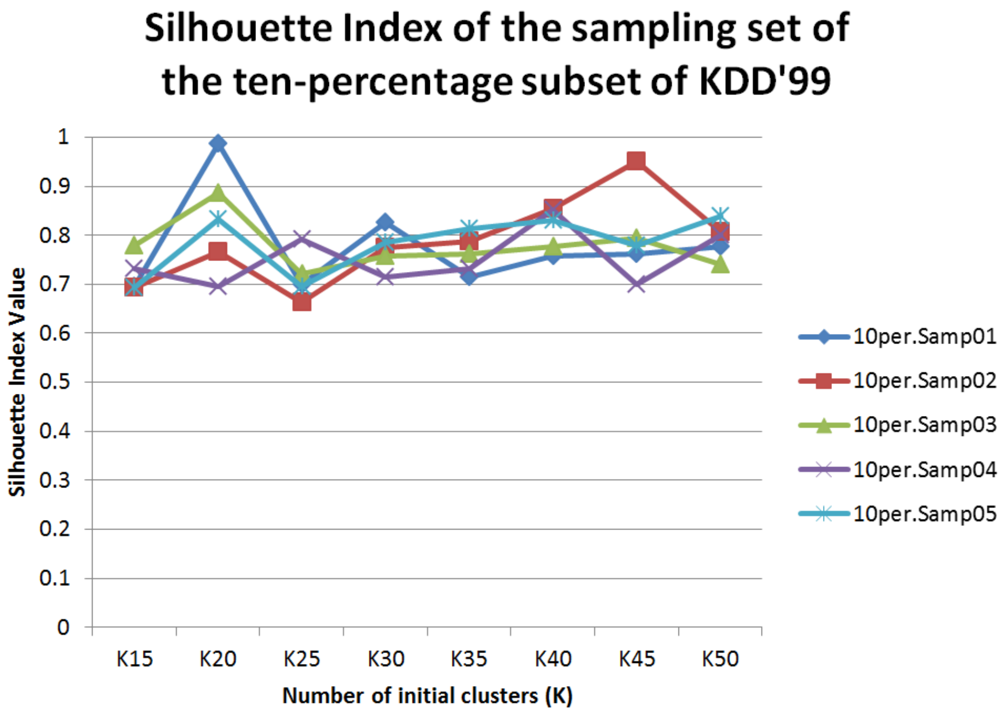


Figure 4.3: The results of the Silhouette index with the various numbers of initial K cluster of sampling set of the ten-percentage subset of KDD’99

According to the Figure 4.3, K20 yields the good result of the silhouette index for the sampling set of the ten-percentage subset of KDD’99. At K20, there are three from five sampling data sets which produce the Silhouette index value greater than 0.80. Therefore, K20 is determined to be an appropriate number of K for the sampling data set of the ten-percentage subset of KDD’99. The Silhouette index result of the ten-percentage subset of KDD’99 also shown in Figure 4.4.

According to the Figure 4.4, K25 and K50 produce the good results of Silhouette index. However, K50 is an excessive number of initial clusters (K). Therefore, K25 is determined to be an appropriate number of K for the ten-percentage subset of KDD’99.

From the Figure 4.2, Figure 4.3 and Figure 4.4, the accuracy graph of the ten-percentage subset of KDD’99 and its sampling sets and the silhouette index value graph of the ten-percentage subset of KDD’99 and its sampling sets are consistent. For the ten-percentage subset of KDD’99, K25 is determined to be an appropriate number of K with the accuracy rate as shown in Figure 4.2. K25 of the results of the Silhouette index as shown in the Figure 4.3 is also determined to be an appropriate number of K. Moreover, K20 is determined to be an appropriate number of K for the sampling sets of the ten-percentage subset of KDD’99 with the accuracy graph and the Silhouette index value graph.

The result of the Silhouette index with the various number of initial clusters (K) graph and the accuracy rate with the various number of initial clusters (K) graph are consistent. Therefore, Silhouette index is applied for the evaluation with other data sets that no labeled for each record. However, KDD’99 data set is the labeled data set. Therefore, the accuracy rate and the various numbers of initial clusters (K) are applied for the evaluations.

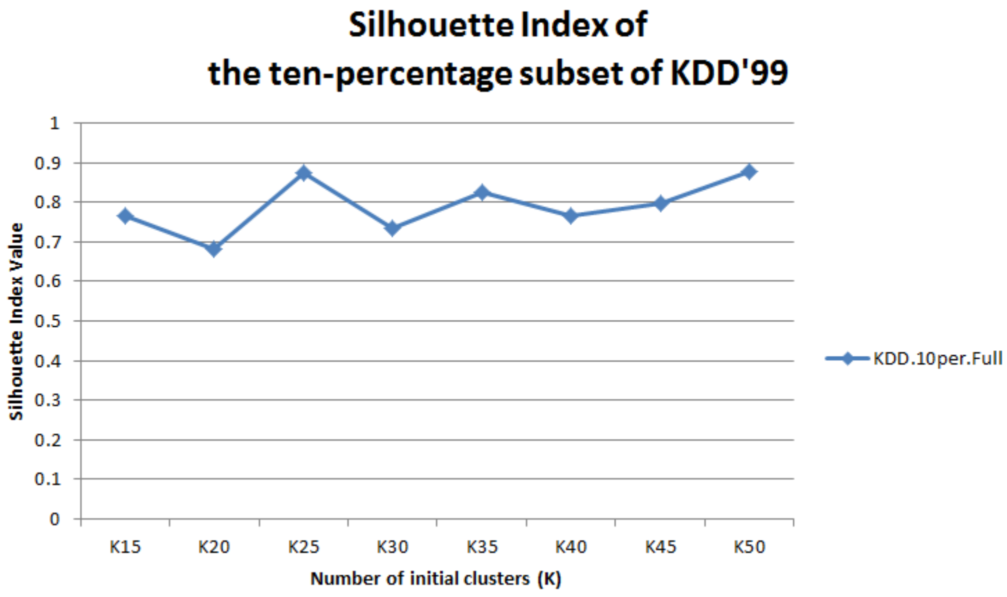


Figure 4.4: The results of the Silhouette index with the various numbers of initial K cluster of the ten-percentage subset of KDD’99

The ROC graph of the ten-percentage subset of KDD'99 is shown in Figure 4.5. It shows the fraction of true positive rate (TPR) and false positive rate (FPR) of various numbers of clusters (K). According to the graph, K5, K10, K15, K20, K25, K30, K35, K40, K45, K50 are selected to plot in the ROC graph. Moreover, the value pairs of TPR and FPR of each selected number of K are shown in Table 4.4 and the ROC graph of these selected numbers of K is shown in Figure 4.5.

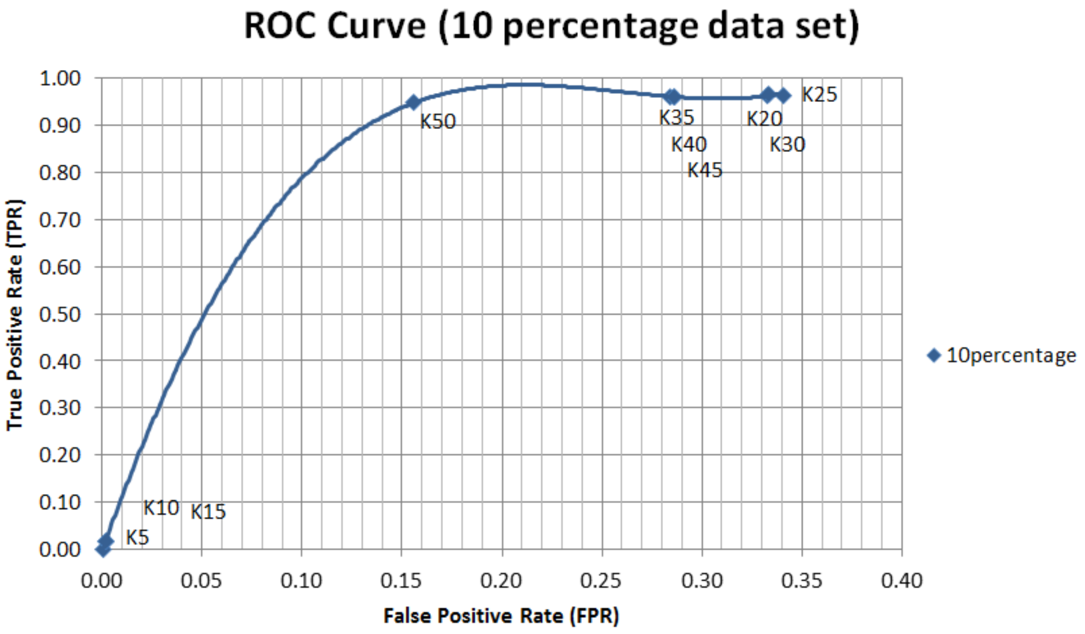


Figure 4.5: ROC curve of the ten-percentage subset of KDD'99

According to the Figure 4.5, K50 produce the best results of true positive rate (TPR) and false positive rate (FPR). However, K35, K40 and K45 are also the proper group of threshold. Nonetheless, these numbers of K are too excessive. Therefore, K20, K25 and K30 are chosen to be a proper group with the adequate number of K. As shown in the graph, these groups have closely value of TPR and FPR. Therefore, Figure 4.6 shows the details information. Figure 4.6 to Figure 4.10 are ROC graph of the 1st sampling data set to 5th sampling data set of the ten-percentage subset of KDD'99 respectively.

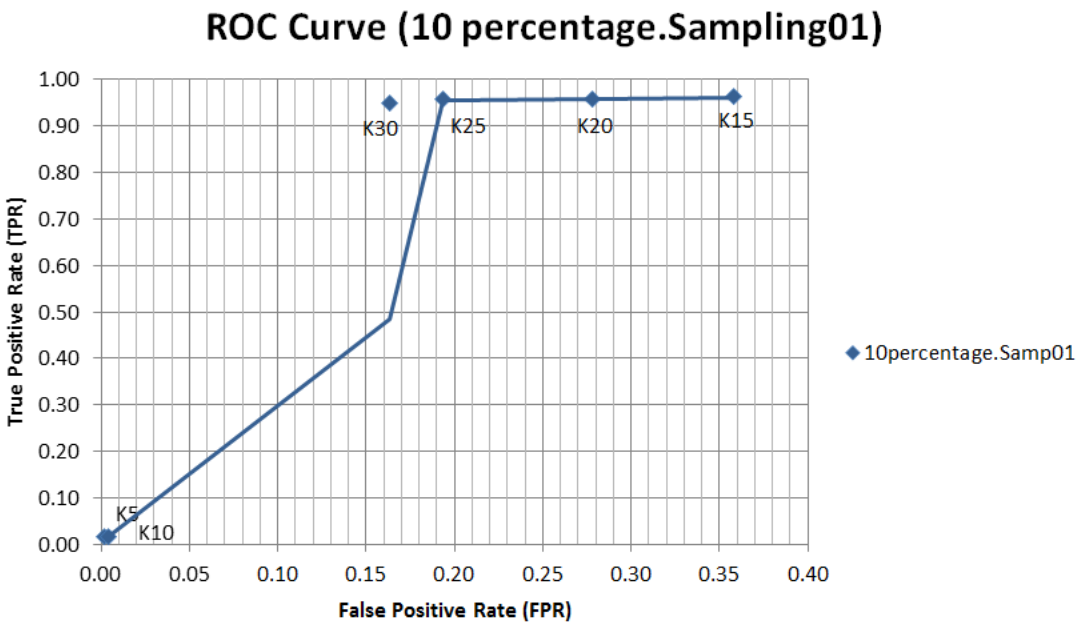


Figure 4.6: ROC curve of the 1st sampling data set of the ten-percentage subset of KDD'99

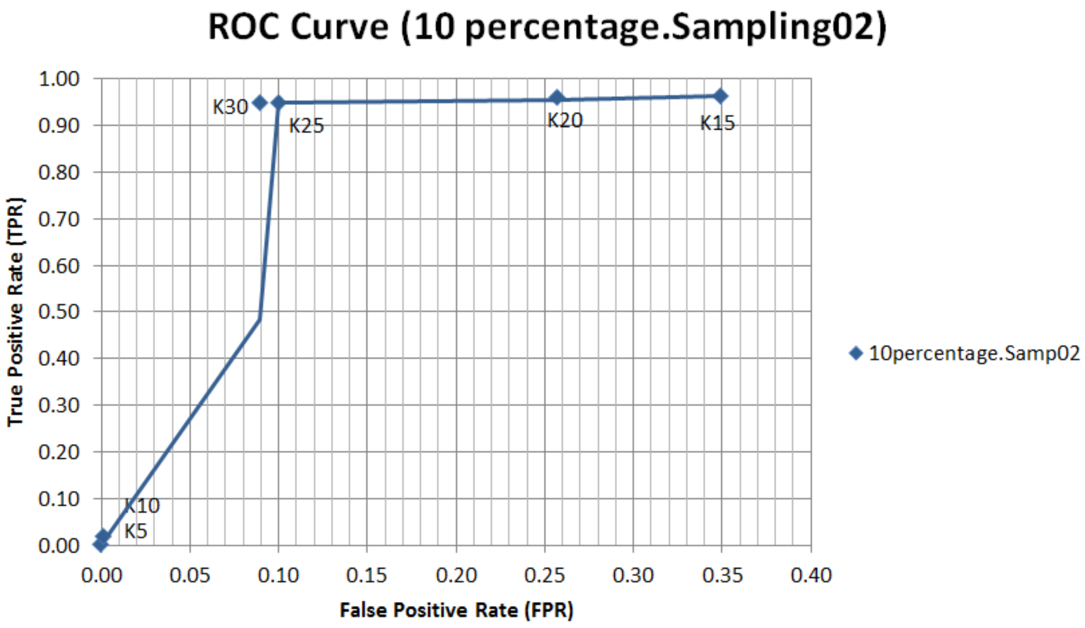


Figure 4.7: ROC curve of the 2nd sampling data set of the ten-percentage subset of KDD'99

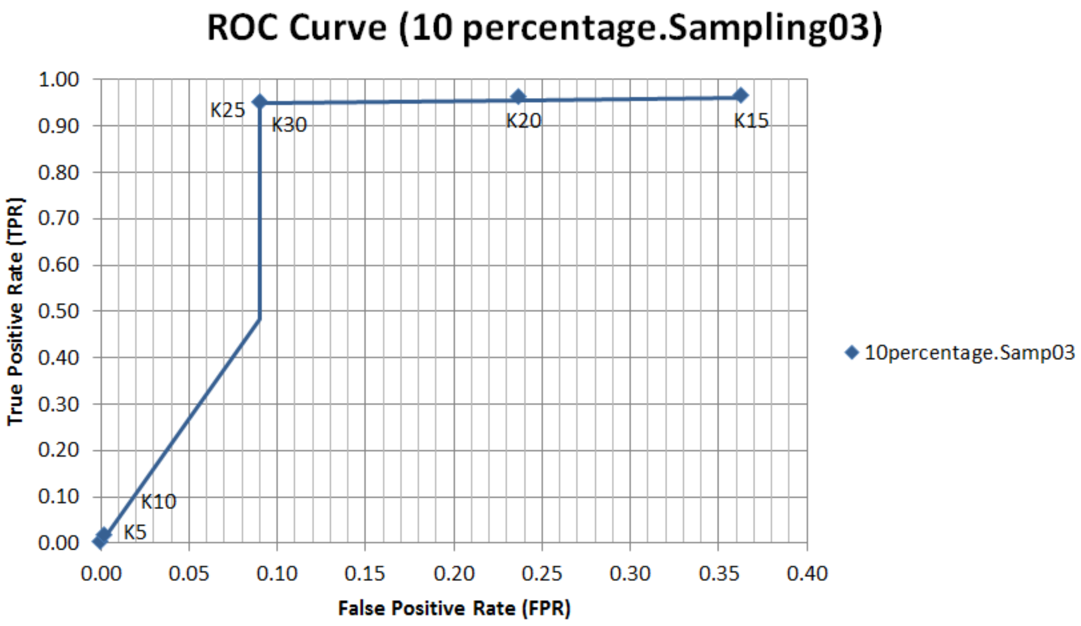


Figure 4.8: ROC curve of the 3rd sampling data set of the ten-percentage subset of KDD’99

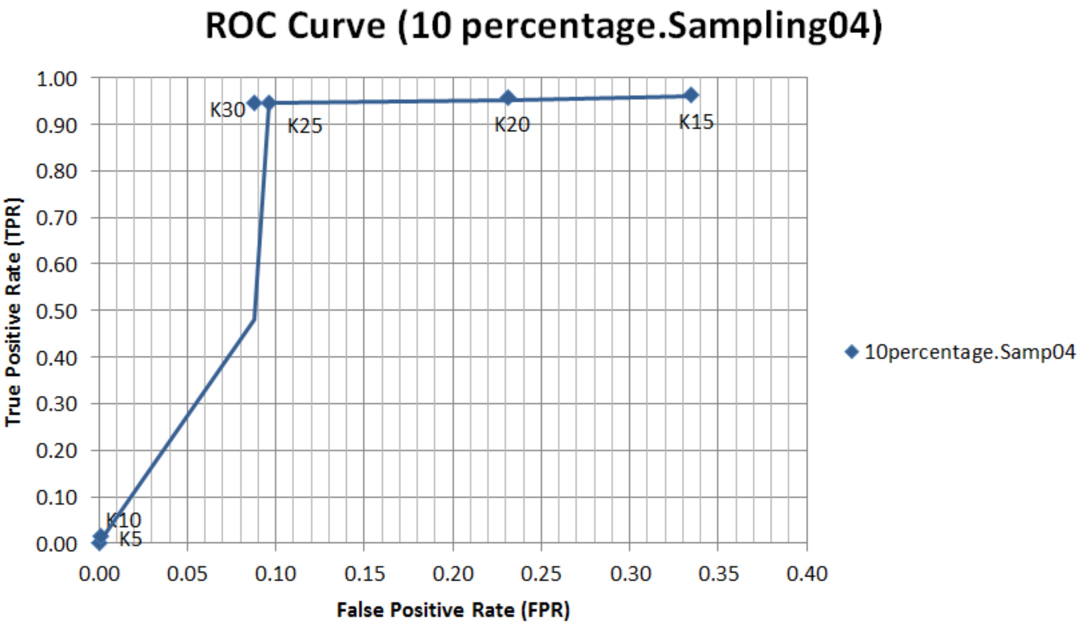


Figure 4.9: ROC curve of the 4th sampling data set of the ten-percentage subset of KDD’99

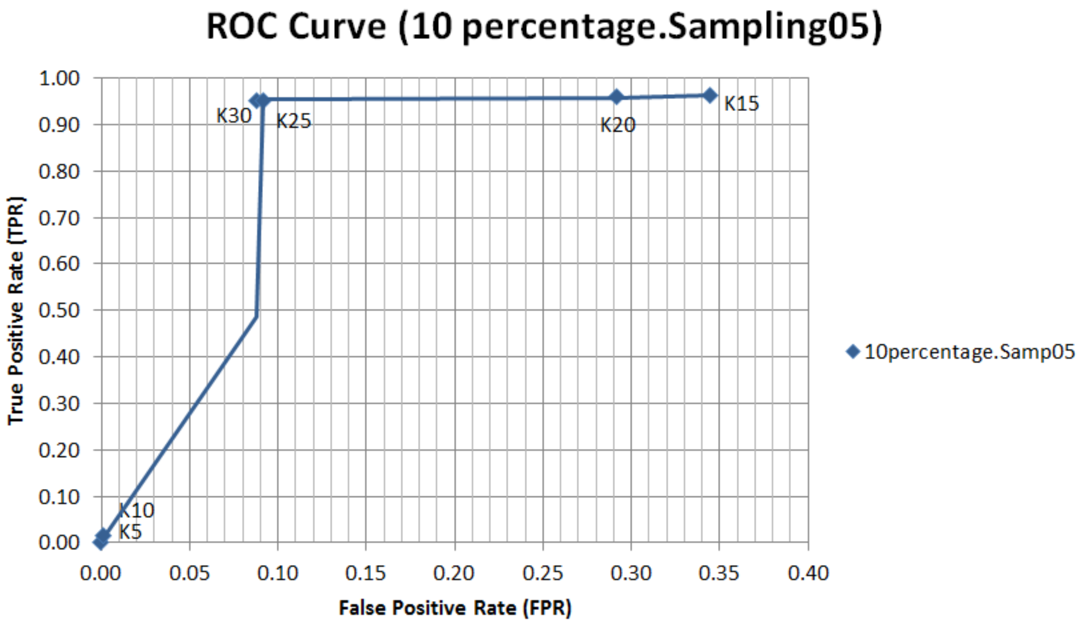


Figure 4.10: ROC curve of the 5th sampling data set of the ten-percentage subset of KDD’99

From the Figure 4.6 to Figure 4.10, they are ROC curve of all sampling data set of the ten-percentage subset of KDD'99. We defined that the maximum proper number of K is not greater than 30. According to the graph, K25 produces the results better than that of K20. Moreover, K25 and K30 closely produce the high rate of TPR with low rate of FPR. With these two numbers of K, K25 and K30, K25 is chosen to be the appropriate number of K, since it is the minimum number of K which yields the high rate of TPR with low rate of FPR.

According to the Table 4.4, K15, K20, K25 and K30 are the appropriate number of K. They produce high rate of TPR and low rate of FPR. However, K15 to K30 does not produce the good results with the ten-percentage subset of KDD’99 data set. Therefore, K25 is determined to be an appropriate number of initial clusters (K) for the ten-percentage subset of KDD’99 data set and its sampling set. Also the results of TNR and FNR of K25 are shown in Table 4.5. At K25, it also produces high rate of TNR and low rate of FNR. So, K25 is determined to be a proper number of K for the data set which not larger than one hundred and fifty thousand records. However, Silhouette index is another evaluation and validation index for clustering with the unlabeled data to indicate the appropriate number of K instead of ROC Curve.

Table 4.4: True Positive Rate (TPR) and False Positive Rate (FPR) of ten-percentage subset of KDD'99 and its sampling data set

K	10percentage		10per (Sampling01)		10per (Sampling02)		10per (Sampling03)		10per (Sampling04)		10per (Sampling05)	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
5	0.00132	0.00034	0.01623	0.00405	0.00138	0.00008	0.00126	0.00000	0.00127	0.00033	0.00137	0.00008
10	0.01783	0.00198	0.01648	0.00215	0.01897	0.00158	0.01662	0.00182	0.01526	0.00124	0.01633	0.00125
15	0.01847	0.00203	0.96425	0.35824	0.96408	0.34898	0.96424	0.36230	0.96364	0.33468	0.96447	0.34435
20	0.96480	0.33265	0.95892	0.27807	0.96107	0.25710	0.95983	0.23659	0.95805	0.23150	0.96098	0.29193
25	0.96470	0.34075	0.95892	0.19336	0.95001	0.09993	0.94862	0.09014	0.94699	0.09601	0.95375	0.09174
30	0.96587	0.33353	0.94942	0.16356	0.94838	0.08938	0.94988	0.09031	0.94559	0.08777	0.95350	0.08824
35	0.96061	0.28578	0.95005	0.09404	0.93055	0.00183	0.94963	0.08616	0.93148	0.00115	0.95350	0.08982
40	0.96059	0.28424	0.94929	0.02956	0.93984	0.00440	0.94875	0.02629	0.94444	0.02794	0.95250	0.03039
45	0.96078	0.28605	0.93902	0.00471	0.93921	0.00440	0.94862	0.02795	0.94495	0.02901	0.94327	0.00509
50	0.94845	0.15598	0.93813	0.00421	0.93670	0.00357	0.94560	0.00862	0.92601	0.00148	0.94278	0.00467

Table 4.5: True Negative Rate (TNR) and False Negative Rate (FNR) of the ten-percentage subset of KDD'99 and its sampling data set

K	10percentage		10per (Sampling01)		10per (Sampling02)		10per (Sampling03)		10per (Sampling04)		10per (Sampling05)	
	TNR	FNR	TNR	FNR	TNR	FNR	TNR	FNR	TNR	FNR	TNR	FNR
5	0.99966	0.99868	0.99595	0.98377	0.99992	0.99862	1.00000	0.99874	0.99967	0.99873	0.99992	0.99863
10	0.99802	0.98217	0.99785	0.98352	0.99842	0.98103	0.99818	0.98338	0.99876	0.98474	0.99875	0.98367
15	0.99797	0.98153	0.64176	0.03575	0.65102	0.03592	0.63770	0.03576	0.66532	0.03636	0.65565	0.03553
20	0.66735	0.03520	0.72193	0.04108	0.74290	0.03893	0.76341	0.04017	0.76850	0.04195	0.70807	0.03902
25	0.65925	0.03530	0.80664	0.04108	0.90007	0.04999	0.90986	0.05138	0.90399	0.05301	0.90826	0.04625
30	0.66647	0.03413	0.83644	0.05058	0.91062	0.05162	0.90969	0.05012	0.91223	0.05441	0.91176	0.04650
35	0.71422	0.03939	0.90596	0.04995	0.99817	0.06945	0.91384	0.05037	0.99885	0.06852	0.91018	0.04650
40	0.71576	0.03941	0.97044	0.05071	0.99560	0.06016	0.97371	0.05125	0.97206	0.05556	0.96961	0.04750
45	0.71395	0.03922	0.99529	0.06098	0.99560	0.06079	0.97205	0.05138	0.97099	0.05505	0.99491	0.05673
50	0.84402	0.05155	0.99579	0.06187	0.99643	0.06330	0.99138	0.05440	0.99852	0.07399	0.99533	0.05722

4.3.1.2. The Full Data Set of KDD’99

The results of accuracy and the number of K for full data set of KDD’99 and its sampling data sets are shown in Figure 4.11.

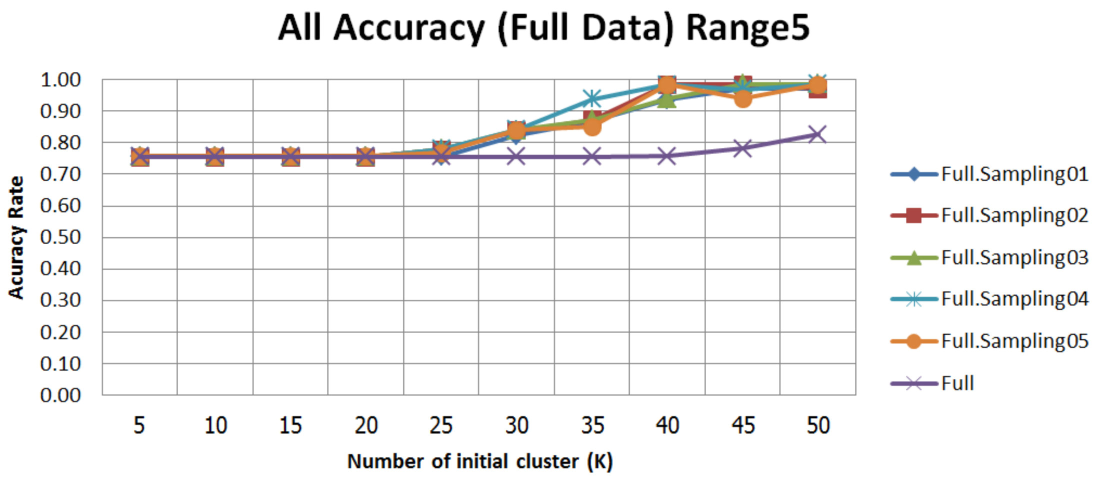


Figure 4.11: The accuracy graph of the full data set of KDD’99 distinct training data set and its sampling data with number of K ranging from 2 to 50 (period of 5)

According to the Figure 4.11, each test data set has approximately the accuracy rate of 0.75 at the beginning. We defined the acceptable rate of accuracy is 0.80 and the proper number of K is not greater than 30.

Table 4.6 and Table 4.7 show the details information of the value pairs of TPR and FPR and the value pairs of TNR and FNR of all of the test data set respectively. According to the Table 4.6 and Table 4.7, we defined the acceptable rate of TPR and TNR is greater than 0.80 and the acceptable rate of FPR and FNR is less than 0.20. Thus, K30 is not adequate to be an appropriate number of K for these test data sets anymore.

Therefore, we determine that the appropriate number of K is greater than 30. According to the Table 4.6 and Table 4.7, K35 is determined to be an appropriate number of K. At 35, all of the sampling data sets have approximately accuracy rate of 0.92. In contrast, a full data set of KDD'99 data sets has approximately the accuracy rate of 0.75.

Table 4.6: True Positive Rate (TPR) and False Positive Rate (FPR) of ten-percentage subset of KDD'99 and its sampling data set

K	Full.Data		Full.Data (Sampling01)		Full.Data (Sampling02)		Full.Data (Sampling03)		Full.Data (Sampling04)		Full.Data (Sampling05)	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
5	0.00003	0.00000	0.00006	0.00000	0.00002	0.00000	0.00004	0.00000	0.00004	0.00000	0.00002	0.00000
10	0.00009	0.00002	0.00006	0.00000	0.00002	0.00000	0.00369	0.00107	0.00369	0.00107	0.00008	0.00001
15	0.00160	0.00008	0.00012	0.00001	0.00374	0.00091	0.00373	0.00102	0.00630	0.00171	0.00008	0.00001
20	0.00228	0.00034	0.00231	0.00034	0.00549	0.00101	0.00267	0.00069	0.00010	0.00001	0.00202	0.00021
25	0.00198	0.00023	0.00231	0.00034	0.96711	0.28298	0.96520	0.28021	0.96520	0.28021	0.96308	0.29592
30	0.00198	0.00016	0.96725	0.22484	0.96227	0.20102	0.96169	0.19821	0.96204	0.19730	0.96418	0.20105
35	0.00198	0.00023	0.96422	0.16247	0.95373	0.15354	0.95373	0.15354	0.95315	0.06758	0.95582	0.18414
40	0.00556	0.00118	0.95591	0.06862	0.94667	0.00251	0.95644	0.06604	0.94667	0.00251	0.94992	0.00455
45	0.96745	0.27760	0.95308	0.02153	0.95012	0.00319	0.94999	0.00294	0.95443	0.02595	0.95483	0.06569
50	0.96757	0.21834	0.95275	0.02391	0.95430	0.02344	0.95002	0.00290	0.95004	0.00291	0.94935	0.00349

Table 4.7: True Negative Rate (TNR) and False Negative Rate (FNR) of the full data set of KDD'99 and its sampling data set

K	Full.Data		Full.Data (Sampling01)		Full.Data (Sampling02)		Full.Data (Sampling03)		Full.Data (Sampling04)		Full.Data (Sampling05)	
	TNR	FNR	TNR	FNR	TNR	FNR	TNR	FNR	TNR	FNR	TNR	FNR
5	1.00000	0.99997	1.00000	0.99994	1.00000	0.99998	1.00000	0.99996	1.00000	0.99996	1.00000	0.99998
10	0.99998	0.99991	1.00000	0.99994	1.00000	0.99998	0.99893	0.99631	0.99893	0.99631	0.99999	0.99992
15	0.99992	0.99840	0.99999	0.99988	0.99909	0.99626	0.99898	0.99627	0.99829	0.99370	0.99999	0.99992
20	0.99966	0.99772	0.99966	0.99769	0.99899	0.99451	0.99931	0.99733	0.99999	0.99990	0.99979	0.99798
25	0.99977	0.99802	0.99966	0.99769	0.71702	0.03289	0.71979	0.03480	0.71979	0.03480	0.70408	0.03692
30	0.99984	0.99802	0.77516	0.03275	0.79898	0.03773	0.80179	0.03831	0.80270	0.03796	0.79895	0.03582
35	0.99977	0.99802	0.83753	0.03578	0.84646	0.04627	0.84646	0.04627	0.93242	0.04685	0.81586	0.04418
40	0.99882	0.99444	0.93138	0.04409	0.99749	0.05333	0.93396	0.04356	0.99749	0.05333	0.99545	0.05008
45	0.72240	0.03255	0.97847	0.04692	0.99681	0.04988	0.99706	0.05001	0.97405	0.04557	0.93431	0.04517
50	0.78166	0.03243	0.97609	0.04725	0.97656	0.04570	0.99710	0.04998	0.99709	0.04996	0.99651	0.05065

According to the Figure 4.11, the trend line of an accuracy rate of the full data set of KDD’99 is quite stable. There is a little raise of curve at the end of the trend line. Therefore, sampling data sets of full data set of KDD’99 are discussed first. For more details, ROC curve of the sampling data sets of full data set of KDD’99 are presented.

The ROC graph of the sampling data sets of the full data set of KDD'99 is shown in Figure 4.12 to Figure 4.16. It shows the fraction of true positive rate (TPR) and false positive rate (FPR) of various numbers of clusters (K). According to the Figure 4.11 and our discussion, we determine that the appropriate number of K for the sampling data sets of the full data set of KDD'99 is 35. Therefore, in this plotting, we select only 7 numbers of K. There are K5, K10, K15, K20, K25, K30 and K35.

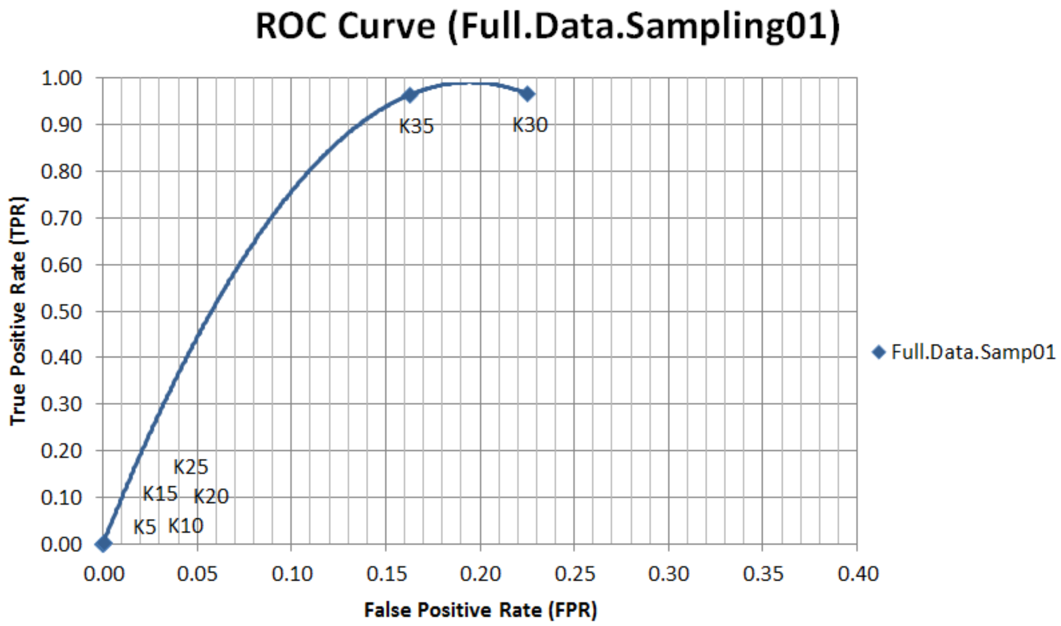


Figure 4.12: ROC curve of the 1st sampling data set of the full data set of KDD’99

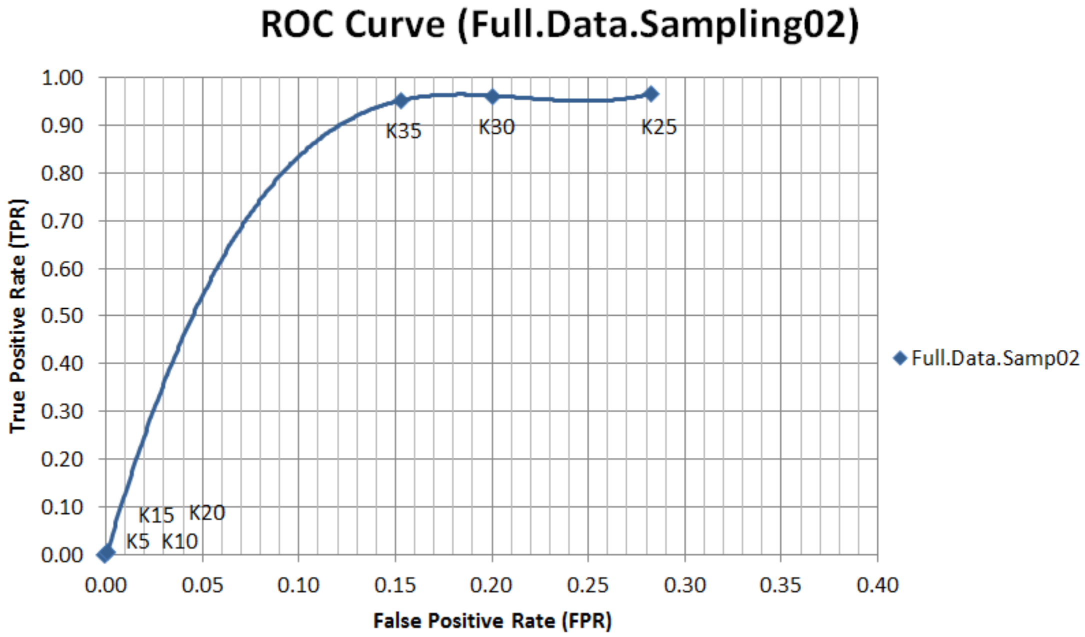


Figure 4.13: ROC curve of the 2nd sampling data set of the full data set of KDD’99

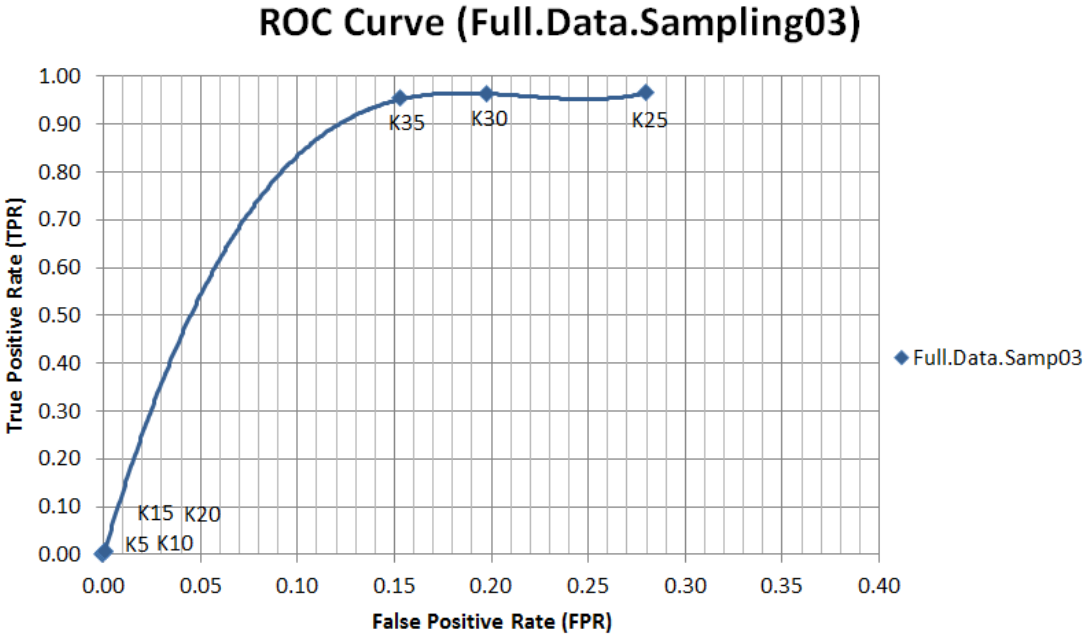


Figure 4.14: ROC curve of the 3rd sampling data set of the full data set of KDD’99

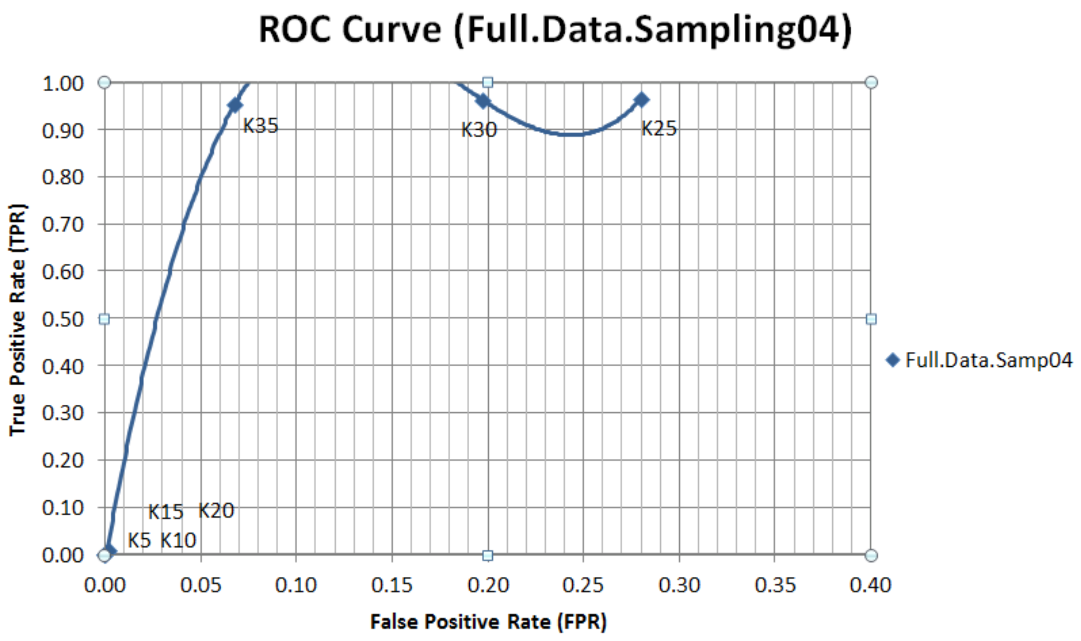


Figure 4.15: ROC curve of the 4th sampling data set of the full data set of KDD'99

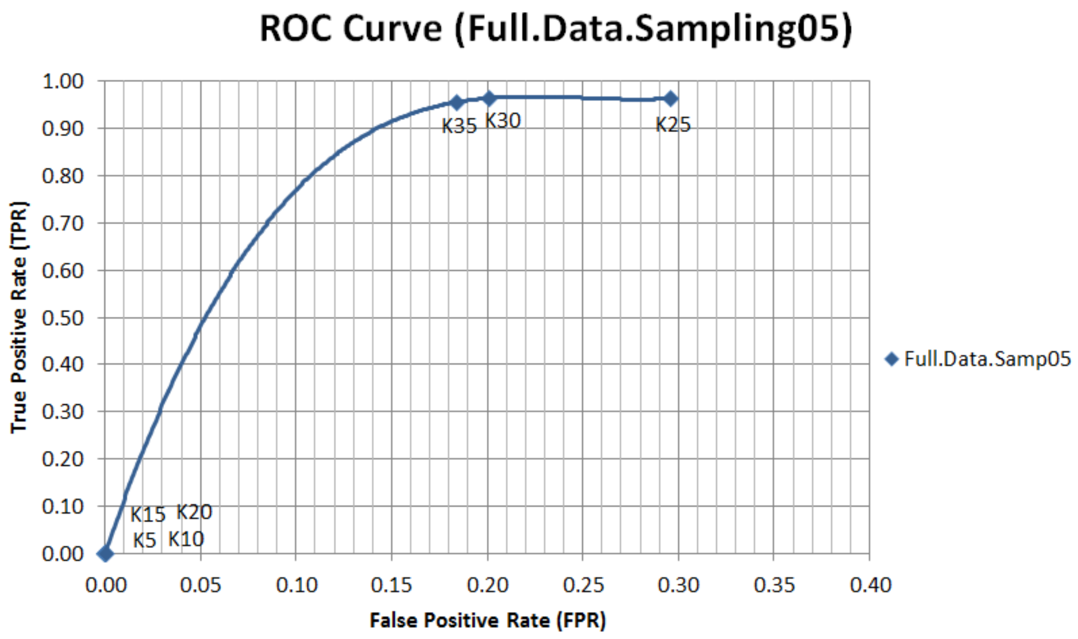


Figure 4.16: ROC curve of the 5th sampling data set of the full data set of KDD'99

According to the Figure 4.12 to Figure 4.16, K35 produces the best results than that of the other number of K. At 35, TPR rate is approximately 0.95 and FPR rate is approximately 0.18 which better than the TPR rate and FPR rate of K25 and K30. Therefore, K35 is determined to be an appropriate number of K for the sampling data set of full data set of KDD'99. Moreover, we determine that K35 is the appropriate number of K for the data set which larger than one hundred and fifty thousand records but not larger than two hundred thousand records.

For the full data set of KDD'99, its trend line of accuracy rate is different from it sampling data set. The ROC curve of the full data set of KDD'99 is shown in Figure 4.17.

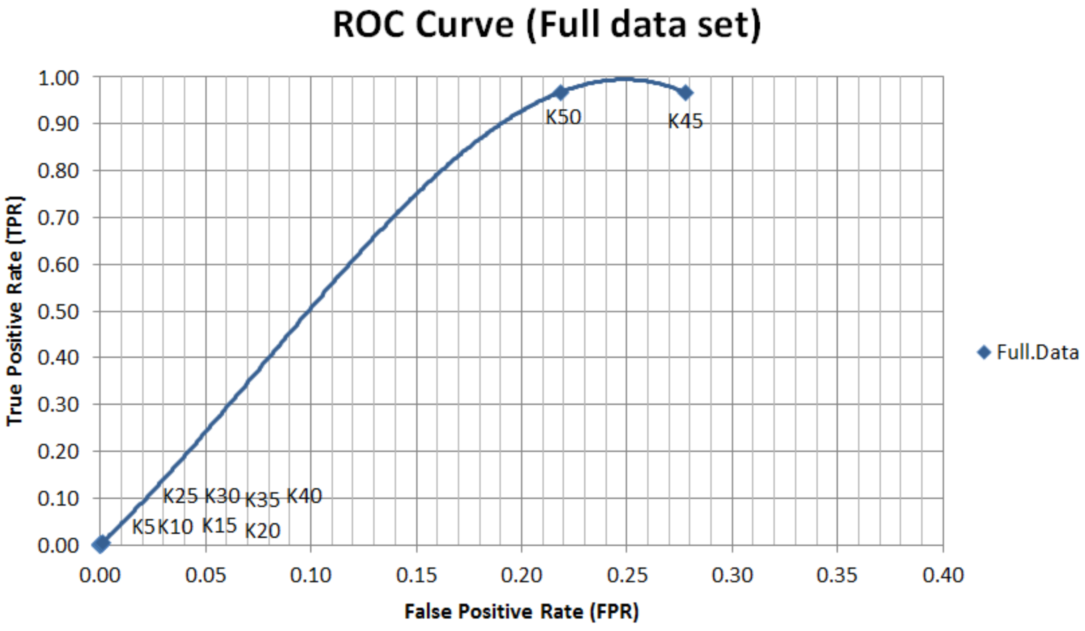


Figure 4.17: ROC curve of the full data set of KDD'99

As shown in the Figure 4.17, K50 is determined to be an appropriate number of K for the full data set of KDD'99 which has approximately one million and seventy five thousand records. At K50, it produces approximately 0.96 of TPR rate with approximately 0.21 of FPR rate. The details of TPR rate and FPR rate of all data set of full data set of KDD'99 is shown in Table 4.6 which shows the value pairs of

TPR and FPR of K5 to K50. The value pairs of TNR and FNR of all these numbers of K of the full data set of KDD'99 and its sampling data set also shown in the Table 4.7.

4.3.1.3. Conclusion of the appropriate number of K

The conclusion of the appropriate number of K is shown in the Table 4.8.

Table 4.8: The conclusion of the appropriate number of K for the various numbers of records

Number of Records	Accuracy	TPR	FPR	TNR	FNR	Number of K
< 150,000	> 0.80	> 0.80	< 0.20	> 0.80	< 0.20	25
150,001 - 200,000	> 0.80	> 0.80	< 0.20	> 0.80	< 0.20	35
1,000,000	> 0.80	> 0.80	< 0.20	> 0.80	< 0.20	50

According to the Table 4.8, the appropriate number of K for the data set which is not larger than one hundred and fifty thousand records is 25. For the data set which larger than one hundred and fifty thousand records but less than two hundred thousand records, K35 is determined to be an appropriate number of K. For the data set which has approximately one million records, the appropriate number of K is 50. For details information, the TPR, TNR, FPR, FNR, accuracy rate and detection rate of the proposed number of K are describes in Table 4.9.

Table 4.9: The details information of the appropriate number of K
and TPR, FPR, TNR, FNR, Accuracy and Detection rate

Data Set	K	TPR	FPR	TNR	FNR	Accuracy	Detection Rate
10percentage	25	0.96470	0.34075	0.65925	0.03530	0.78042	0.65055
10per.Samp.01	25	0.95892	0.19336	0.80664	0.04108	0.86670	0.76358
10per.Samp.02	25	0.95001	0.09993	0.90007	0.04999	0.91995	0.86278
10per.Samp.03	25	0.94862	0.09014	0.90986	0.05138	0.92525	0.87390
10per.Samp.04	25	0.94699	0.09601	0.90399	0.05301	0.92090	0.86476
10per.Samp.05	25	0.95375	0.09174	0.90826	0.04625	0.92650	0.87439
Full.Data	50	0.96757	0.21834	0.78166	0.03243	0.88545	0.79994
Full.Samp.01	35	0.96422	0.16247	0.83753	0.03578	0.92335	0.86806
Full.Samp.02	35	0.95373	0.15354	0.84646	0.04627	0.97125	0.99704
Full.Samp.03	35	0.95373	0.15354	0.84646	0.04627	0.92805	0.87890
Full.Samp.04	35	0.95315	0.06758	0.93242	0.04685	0.97235	0.99809
Full.Samp.05	35	0.95582	0.18414	0.81586	0.04418	0.92755	0.87666

According to the Table 4.9, the TPR rate, FPR rate, TNR rate, FNR rate, accuracy rate and detection rate of all the test data set is quite precisely to the defined acceptable rate as shown in the Table 4.8. With the proposed number of K with the various size number of data set, there are some values which less than the defined acceptable rate. However, most of the values with these proposed numbers of K are exceed the defined rate.

4.3.2. The Appropriate Number of Iteration

In this part, the appropriate number of iteration is discussed. The ten-percentage subset of KDD'99 and the full data set of KDD'99 are used. For the ten-percentage subset of KDD'99, it completed its job in 102 iterations. For the full data set of KDD'99, it completed its job in 300 iterations which is the maximum defined iteration. The accuracy rate with each number of iteration graphs is shown in Figure 4.18.

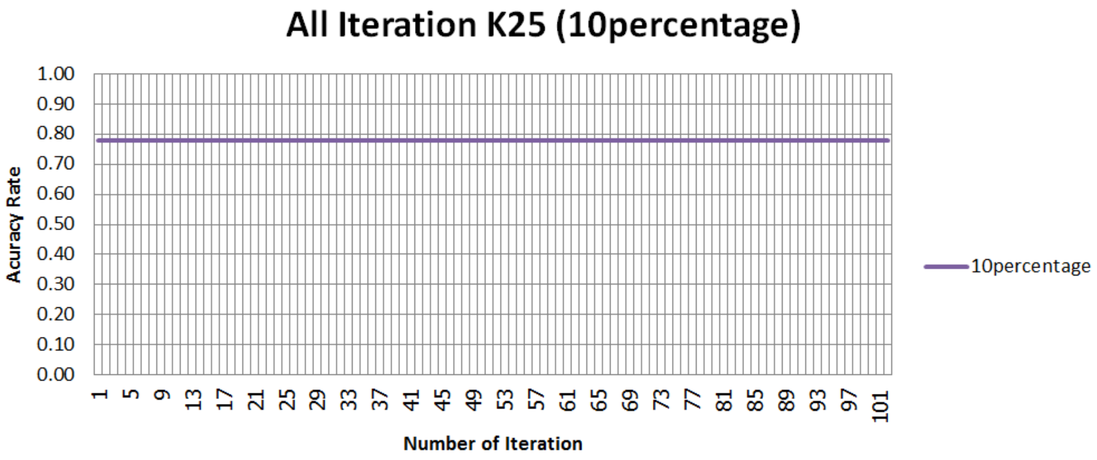


Figure 4.18: The accuracy rate with each number of iteration

of the ten-percentage subset of KDD'99 with K25

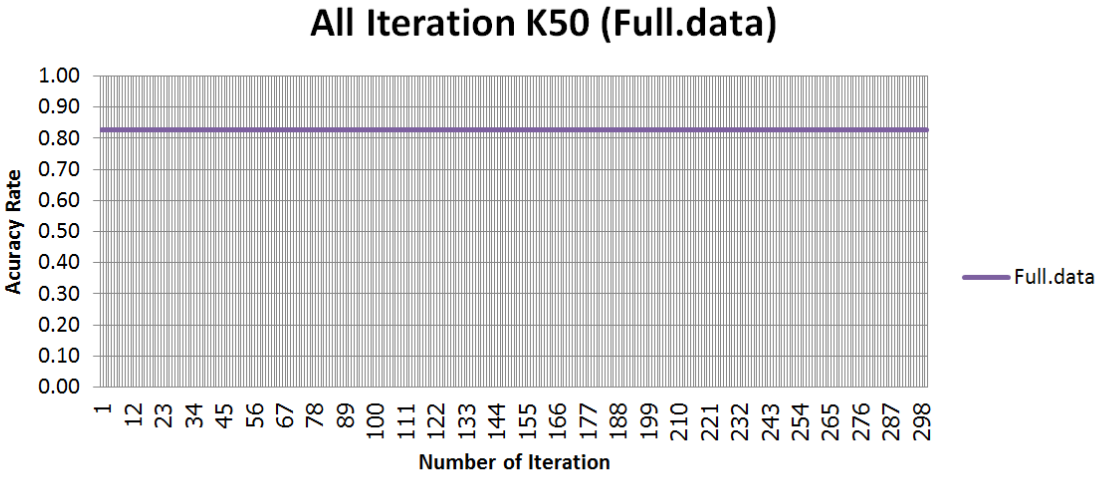


Figure 4.19: The accuracy rate with each number of iteration of the full data set of KDD'99 with K50

According to the Figure 4.18 and Figure 4.19, the accuracy rate of each number of iteration is equals. It shows that all iterations produce the same results. One of the reasons of these characteristics of the graph is that K-means algorithm based on Apache Mahout has the specific process to generate the initial K centroids. This specific process is Canopy clustering. Canopy clustering is first applied to partition data into groups. The center points of each group from the Canopy clustering are used to be the initial K centroids of the K-means algorithm. Therefore, ten iterations are enough to be an appropriate number of iteration. Moreover, the default value of maximum number of iteration in Apache Mahout is ten iterations. Thus, the appropriate number of iteration of our implementation is ten iterations.

4.4 Association Rule Mining Using Parallel FP-Growth algorithm

In this Section, Parallel FP-Growth algorithm which is an association rule mining is applied to retrieve the normal profiles from the log files. According to the Table 4.3, most of the test data set has the normal records more than 60% of all records. Moreover, the defined threshold which is used in [2] for the association rule mining to generate firewall rules is 70%. Therefore, we defined that the minimum threshold for the Parallel FP-Growth in our implementation is 70%.

In this process, the ten-percentage subset of KDD'99 and the full data set of KDD'99 are used. First, Parallel FP-Growth algorithm is applied to retrieve the rules which their relations occur more than the defined minimum threshold which is 70%. Then, all the rules are classified by the number of attributes. After that, each class of rules is used as the normal profiled of the system. Finally, the TPR, FPR, TNR, FNR, accuracy rate and detection rate are evaluated.

After this process, the ranges of number of attribute which is an appropriate rule are proposed. First, the ten-percentage subset of KDD'99 with the number of K at 25 is discussed. Then, the full data set of KDD'99 with the number of K at 50 is shown.

The appropriate range of number of attribute which is an appropriate rule is discussed with the ROC curve. We defined the acceptable rate of TPR and TNR is greater than 0.80 and the acceptable rate of TNR and FNR is less than 0.20. The ROC curve of the ten-percentage subset of KDD'99 with various number of attribute is shown in Figure 4.20. In this plotting, we select only 9 numbers of attribute lengths with the period of four. They are A4, A8, A12, A16, A20, A24, A28, A32 and A36.

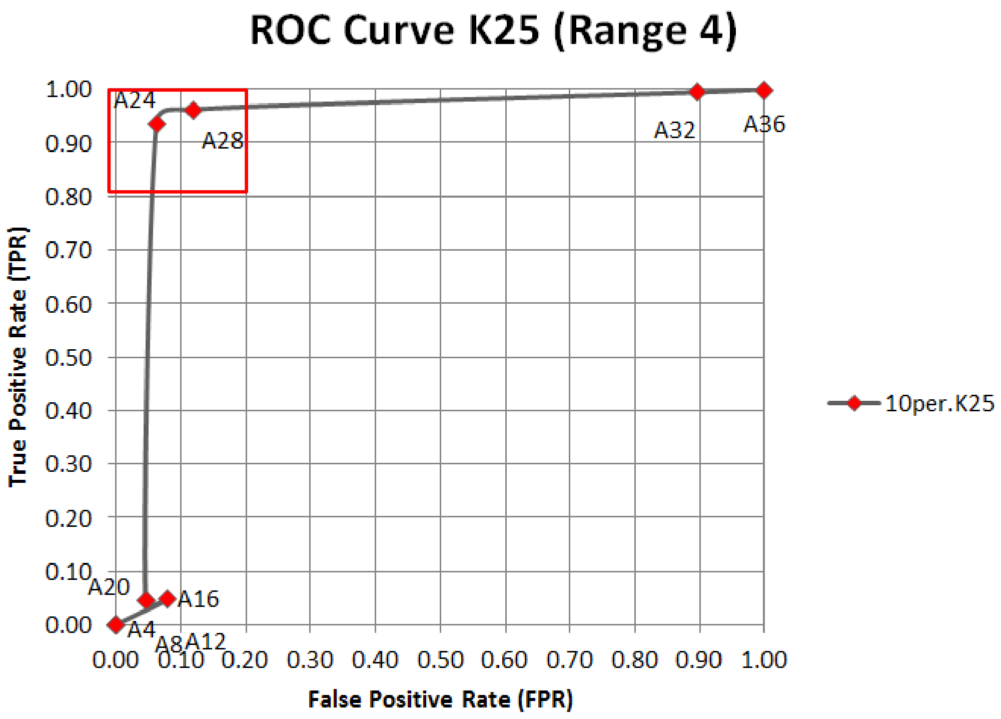


Figure 4.20: The ROC curve of the ten-percentage subset of KDD'99

with the number of K at 25 (period of 4)

As shown in the Figure 4.20, A24 and A28 are the remarkable points. For A24, Its TPR and FPR rate is 0.93 and 0.06 respectively. For A28, it's TPR and FPR rate is 0.96 and 0.11 respectively. The TPR rate of A24 and A28 is greater than the defined acceptable rate which is 0.80. The FPR rate of A24 and A28 are also less than the defined acceptable rate which is 0.20.

The ROC curve is plotted by the fraction of true positive rate (TPR) and false positive rate (FPR). It is used to depict the performance of the system for the intrusion detection. In this process, the normal profiles of the system are proposed. Therefore, the reverse ROC curve is proposed to depict the performance of the system for the normal detection. The reverse ROC curve is plotted by the fraction of true negative rate (TNR) and false negative rate (FNR). The reverse ROC curve of the ten-percentage subset of KDD'99 with the number of K at 25 is shown in Figure 4.21.

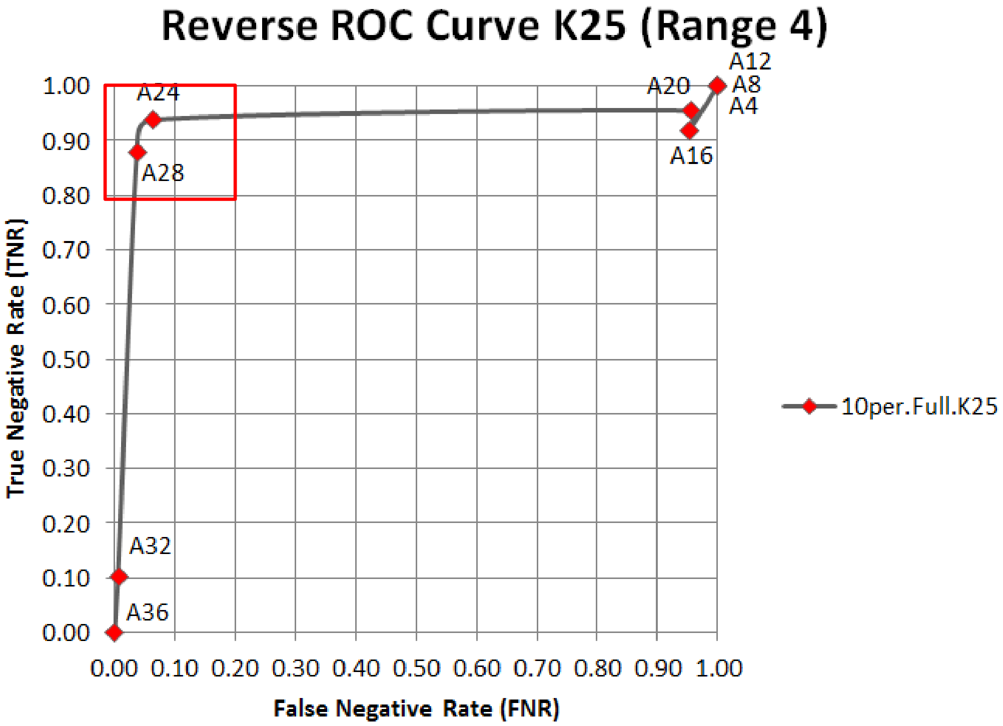


Figure 4.21: The reverse ROC curve of the ten-percentage subset of KDD’99 with the number of K at 25 (period of 4)

As shown in the Figure 4.21, The TNR and FNR rate of A24 and A28 are also in the range of the defined acceptable rate. Therefore, the range of number of attribute from 24 to 28 (A24 and A28) is determined to be an appropriate range of number of attribute which is used to be the normal profiles. For details information, the ROC curve and reverse ROC curve of the range of number of attributes from 22 to 30 are shown in Figure 4.22 and Figure 4.23.

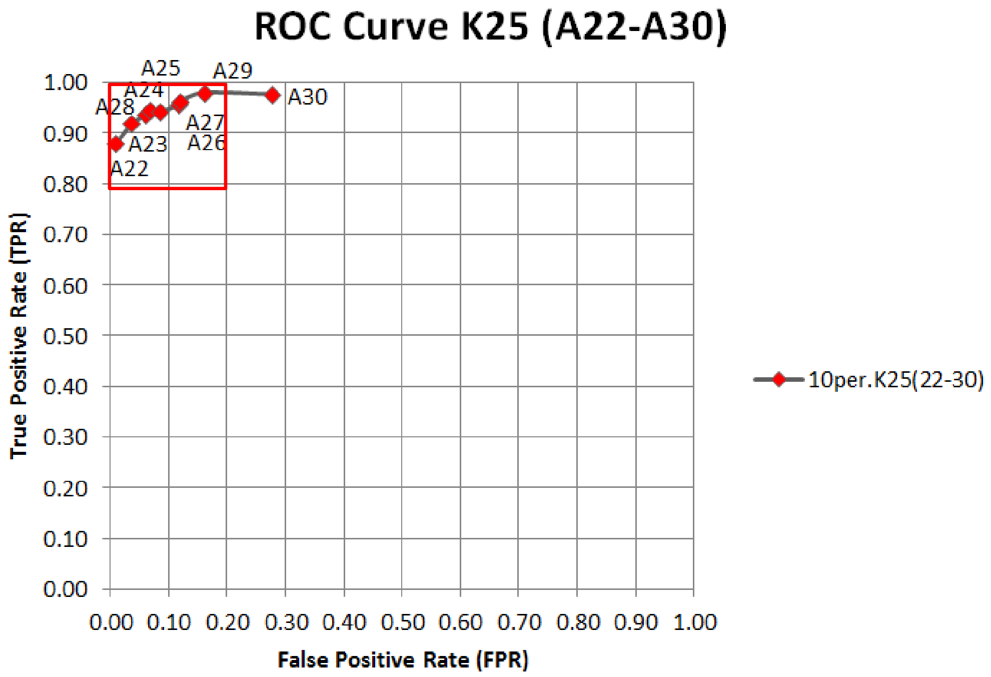


Figure 4.22: The ROC curve of the ten-percentage subset of KDD'99

with the number of K at 25 (A22 – A30)

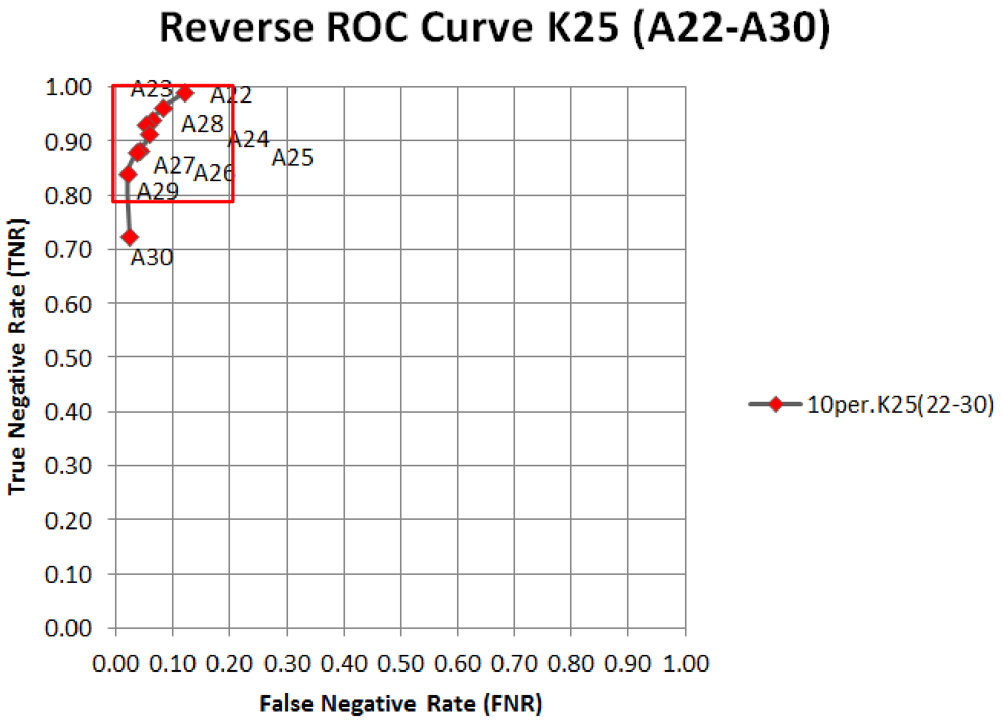


Figure 4.23: The reverse ROC curve of the ten-percentage subset of KDD'99

with the number of K at 25 (A22 – A30)

According to the Figure 4.22 and Figure 4.23, A22 to A29 are determined to be an appropriate range of number of attribute. However, this range of number of attribute is an appropriate range of number of attribute for the ten-percentage subset of KDD'99. Therefore, the ROC curve and the reverse ROC curve of the full data set of KDD'99 with the number of K at 50 are shown in Figure 4.24 and Figure 4.25 respectively.

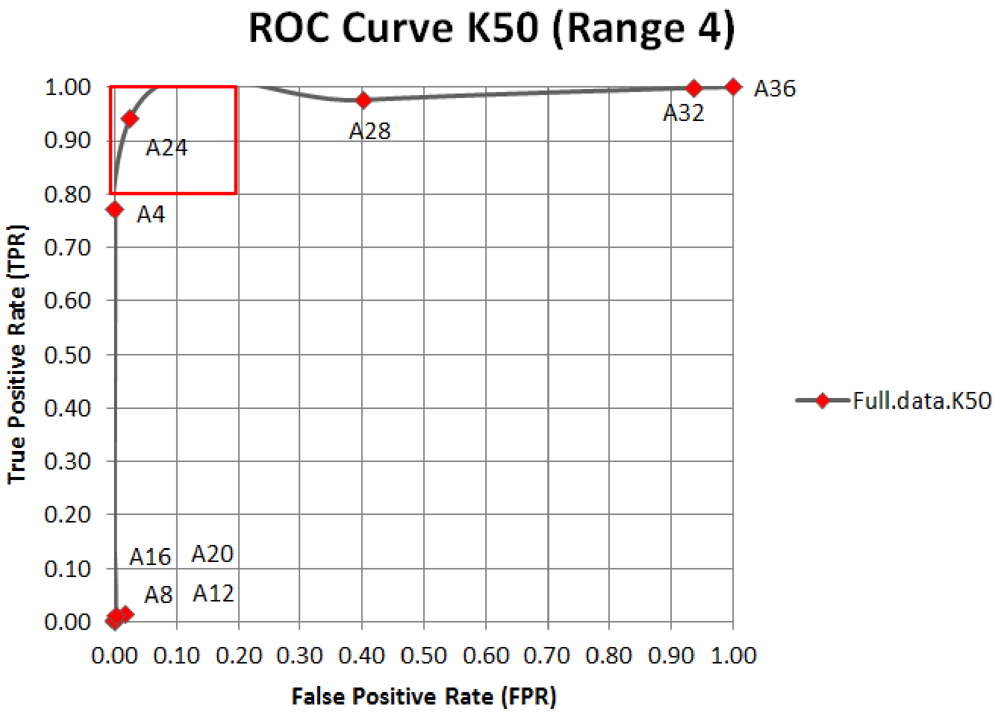


Figure 4.24: The ROC curve of the full data set of KDD'99 with the number of K at 50 (period of 4)

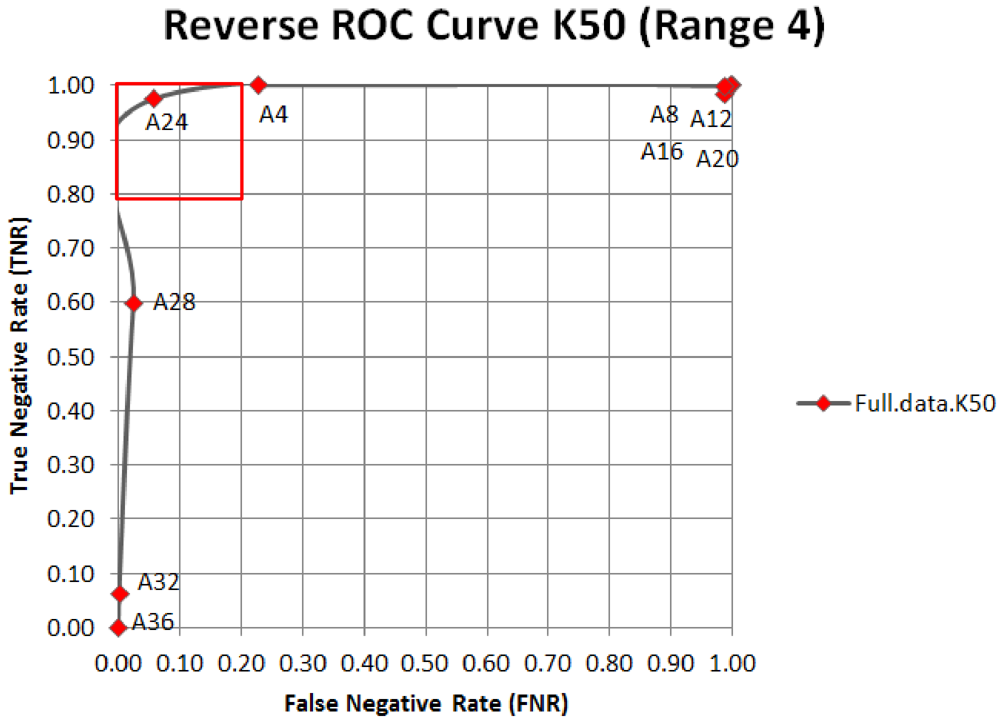


Figure 4.25: The reverse ROC curve of the full data set of KDD'99 with the number of K at 50 (period of 4)

According to the Figure 4.24 and Figure 4.25, only A24 is an appropriate number of attribute to generate the normal profiles. Therefore, the ROC curve and the reverse ROC curve of the full data set of KDD'99 with the number of K at 50 with the range of attribute from 22 to 30 are shown in the Figure 4.26 and Figure 4.27.

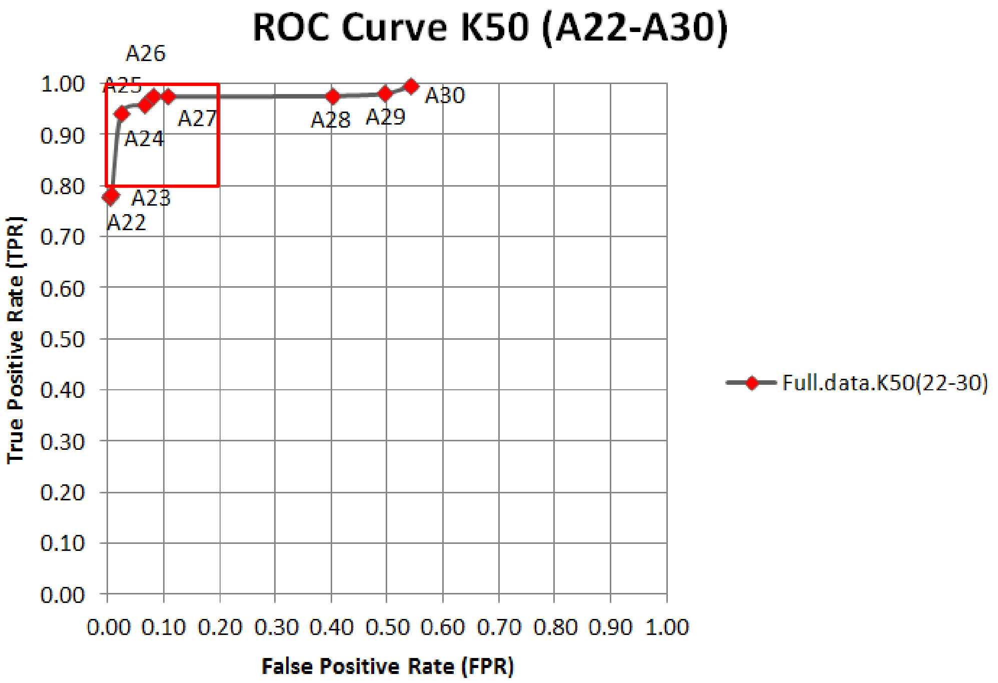


Figure 4.26: The ROC curve of the full data set of KDD'99 with the number of K at 50 (A22 – A30)

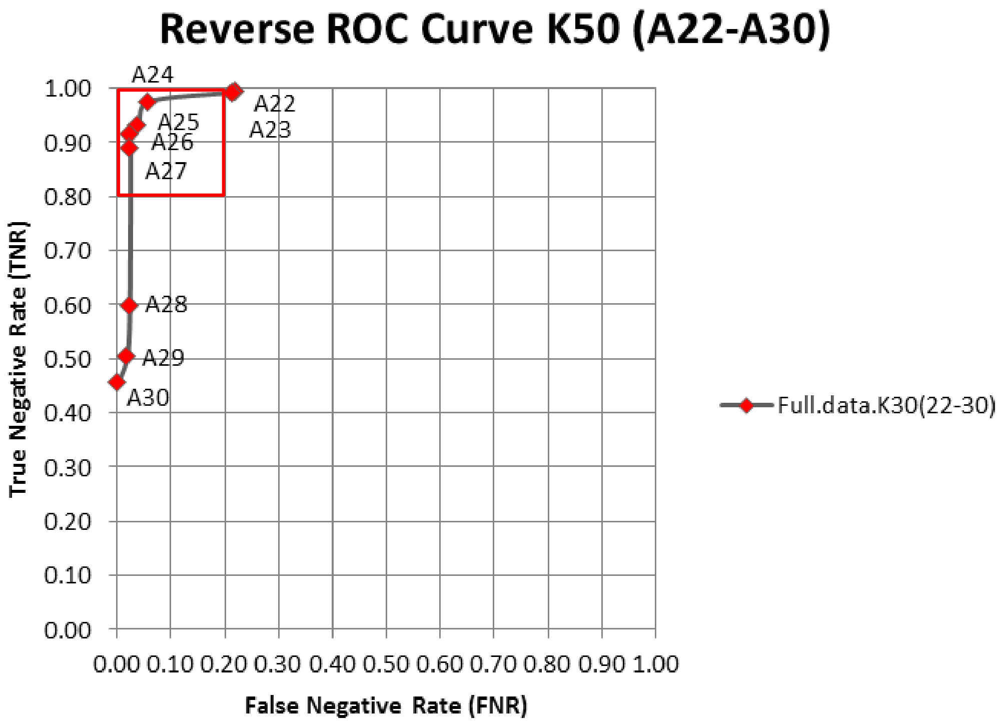


Figure 4.27: The reverse ROC curve of the full data set of KDD'99

with the number of K at 50 (A22 – A30)

According to the Figure 4.26 and Figure 4.27, A24, A25, A26 and A27 are the appropriate group for the range of number of attribute to generate normal profiles. With the previous discussion, we proposed that A22 to A29 are the proper range of number of attributes to generate normal profiles. With these two sets of range of number of attributes, A24, A25. A26 and A27 are the intersection ranges. Therefore, A24, A25. A26 and A27 are determined to be the proper number of attributes to generate normal profiles. In other words, the appropriate number of attributes to generate normal profiles is in the range of 58% to 65% of the number of attributes in each record.

After the proper number of attribute is determined, we evaluate these numbers with the TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate. First, the normal profiles are created from the rule which the number of attribute is in the range of 58% to 65% of all the number of attribute in each record. Then, the input data is compared to the proposed normal profiles. After that, the data set is divided into two groups. They are the normal group and intrusion group. If the record is matched to propose normal profiles, it will be labeled as a normal record. In contrast, if the record is not matched to the proposed normal profiles, it will be labeled as an intrusion record. Finally, confusion matrix is calculated.

The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of the ten-percentage subset of KDD’99 for number of attribute from 4 (A4) to 36 (A36) with the period of four is shown in Table 4.10. In this plotting, we select only 9 numbers of attribute lengths. There are A4, A8, A12, A16, A20, A24, A32 and A36. Moreover, The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of the ten-percentage subset of KDD’99 for number of attribute from 22 (A22) to 30 (A30) is shown in Table 4.11.

As shown in the Table 4.10 and Table 4.11, they also indicate that the appropriate range of number of attribute is between 22 and 29. Their TPR rate, TNR rate and accuracy rate of these number which are in the proposed range are greater than 0.80. Also the FPR rate and FNR rate are less than 0.20. The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of the full data set of KDD’99 are also shown in Table 4.12 and Table 4.13.

Table 4.10: The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of each number of attribute of the ten-percentage subset of KDD'99 with K25 (period of 4)

Number of Attribute	4	8	12	16	20	24	28	32	36
True Positive Rate (TPR)	0.00002	0.00002	0.00038	0.04794	0.04465	0.93687	0.96248	0.99420	0.99969
False Positive Rate (FPR)	0.00000	0.00000	0.00001	0.08002	0.04512	0.06201	0.11958	0.89718	1.00000
True Negative Rate (TNR)	1.00000	1.00000	0.99999	0.91998	0.95488	0.93799	0.88042	0.10282	0.00000
False Negative Rate (FNR)	0.99998	0.99998	0.99962	0.95206	0.95535	0.06313	0.03752	0.00580	0.00031
Accuracy	0.60330	0.60330	0.60344	0.57404	0.59379	0.93755	0.91297	0.45644	0.39658

Table 4.11: The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of each number of attribute of the ten-percentage subset of KDD'99 with K25 (A22 to A30)

Number of Attribute	22	23	24	25	26	27	28	29	30
True Positive Rate (TPR)	0.87985	0.91817	0.93687	0.94582	0.94175	0.95637	0.96248	0.97998	0.97652
False Positive Rate (FPR)	0.00969	0.03714	0.06201	0.06838	0.08635	0.11656	0.11958	0.16083	0.27794
True Negative Rate (TNR)	0.99031	0.96286	0.93799	0.93162	0.91365	0.88344	0.88042	0.83917	0.72206
False Negative Rate (FNR)	0.12015	0.08183	0.06313	0.05418	0.05825	0.04363	0.03752	0.02002	0.02348
Accuracy	0.94649	0.94513	0.93755	0.93725	0.92480	0.91237	0.91297	0.89503	0.82300

Table 4.12: The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of each number of attribute of the full data set of KDD’99 with K50 (period of 4)

Number of Attribute	4	8	12	16	20	24	28	32	36
True Positive Rate (TPR)	0.77282	0.00008	0.01420	0.00450	0.01301	0.94174	0.97597	0.99912	0.99999
False Positive Rate (FPR)	0.00002	0.00001	0.01555	0.00002	0.00075	0.02442	0.40202	0.93747	1.00000
True Negative Rate (TNR)	0.99998	0.99999	0.98445	0.99998	0.99925	0.97558	0.59798	0.06253	0.00000
False Negative Rate (FNR)	0.22718	0.99992	0.98580	0.99550	0.98699	0.05826	0.02403	0.00088	0.00001
Accuracy	0.94458	0.75612	0.74782	0.75719	0.75872	0.96732	0.69017	0.29095	0.24389

Table 4.13: The TPR rate, FPR rate, TNR rate, FNR rate and accuracy rate of each number of attribute of the full data set of KDD’99 with K50 (A22 to A30)

Number of Attribute	22	23	24	25	26	27	28	29	30
True Positive Rate (TPR)	0.77819	0.78473	0.94174	0.95966	0.97545	0.97443	0.97597	0.98029	0.99649
False Positive Rate (FPR)	0.00436	0.00742	0.02442	0.06550	0.08227	0.10809	0.40202	0.49505	0.54272
True Negative Rate (TNR)	0.99564	0.99258	0.97558	0.93450	0.91773	0.89191	0.59798	0.50495	0.45728
False Negative Rate (FNR)	0.22181	0.21527	0.05826	0.04034	0.02455	0.02557	0.02403	0.01971	0.00351
Accuracy	0.94261	0.94189	0.96732	0.94063	0.93180	0.91203	0.69017	0.62088	0.58879

According to the Table 4.12 and Table 4.13, the results also indicate that the appropriate range of number of attributes is between 22 and 27. Their TPR rate, TNR rate and accuracy rate of these number which are in the proposed range are greater than 0.80. Also the FPR rate and FNR rate are less than 0.20.

In conclusion, the appropriate range of number of attributes to generate normal profiles is in the range of 58% to 65% of all the number of attribute in each record. According to the experiments, the proposed ranges of number of attribute given the good enough normal profiles. With our evaluation using confusion matrix, the proposed normal profiles with the proposed range of number of attribute produce high rate of TPR, TNR and accuracy with the low rate of FPR and FNR as shown in the Table 4.14.

Table 4.14: The confusion matrix after the normal patterns are removed by the normal profiles of the full data set of KDD'99 with K50

Matrix	K50
True Positive Rate (TPR)	0.94059
False Positive Rate (FPR)	0.02399
True Negative Rate (TNR)	0.97600
False Negative Rate (FNR)	0.05940
Positive Predictive Value (PPV)	0.92670
Negative Predictive Value (NPV)	0.98074
False Discovery Rate (FDR)	0.07329
Accuracy	0.96736

According to the Table 4.14, the true positive rate is approximately 0.94 with the very low false positive rate. The true negative rate, detection rate and accuracy are greater than 0.95 and the false negative rate and false discovery rate are less than 0.10. Therefore, the proposed normal profiles produce the rather precise results.

4.5 The 2nd Clustering using K-Means algorithm

In this Section, K-Means algorithm is applied to partition the remaining data. From the previous process, the normal profiles are created. Therefore, the normal records are removed from the data set. Thus, the remaining data are the suspect records to be an intrusion. Hence, these remaining data will be partitioned into groups.

After the partitioning, the largest cluster is labeled to be an intrusion cluster. If any cluster has a member greater than one-fourth [1] of the largest cluster, that cluster will be labeled to be an intrusion cluster too. Other clusters will be labeled to be an outlier. Outlier clusters are not included to analyze in the next process.

4.6 The 2nd Association Rule Mining Using Parallel FP-Growth algorithm

In this Section, the clusters which are labeled to be an intrusion are analyzed. Parallel FP-Growth using Apache Mahout is applied to discover the characteristics of these intrusions. The minimum threshold which is the percentage of the minimum occurrence of the relation in the data set is 70% [2]. Therefore, the patterns of relation which frequently occur over 70% are generated. The pattern rules which have the range of 58% to 65% of all the number of attribute in each record are determined to be the characteristics of the suspect intrusion cluster.

After this process, the characteristics of the suspect intrusion are proposed. These characteristics will be examined by the security experts to analyze that which type of attacks they are. Finally, the new knowledge is provided by the analysis of the security expert to support firewall rules.

4.7 Conclusion of the Implementation

Table 4.15: The conclusion of the implementation (K, iterations, percentage of number of attribute)

Number of Records	K	Iterations	Percentage of number of Attribute	
			Normal Profiles	Attacks
< 150,000	25	10	58% - 65%	58% - 65%
150,001 - 200,000	35	10	58% - 65%	58% - 65%
1,000,000	50	10	58% - 65%	58% - 65%

The conclusion of all our implementation is shown in Table 4.15. According to the Table 4.15, the appropriate number of K for the data set which the number of records is less than one hundred and fifty thousand records is 25. For the data set which the number of records are larger than one hundred and fifty thousand records but less than two hundred thousand records, K35 is the proper number of K. For the one million records data set, the appropriate number of K is 50. Moreover, ten iterations are enough for the clustering using K-Means algorithm based on Apache Mahout. The percentage of number of attribute which yields the good normal profiles and intrusion characteristics is between 58% and 65%.

CHAPTER V
EXPERIMENTS

In this chapter, we analyze and evaluate the performance of our implementations. The ten-percentage subset of the test set of KDD’99 is used for the evaluation. Section 5.1 describes the environments of our experiments. Then, the details of the ten-percentage subset of the test set of KDD’99 are shown in Section 5.2. The results of our experiments are discussed in Section 5.3. The results of our system and other system are compared in Section 5.4. The last Section is the conclusion of our experiments.

5.1 Experimental Environments

In our experiments, we use five machines which powered by the following specification.

- Intel(R) Core(TM) i5-2500 CPU @ 3.30GHz
- 4 GB of RAM
- 500 GB of storage

Each node of machine is running the Apache Hadoop 0.20.2-cdh3u5 software distributed by Cloudera on the Ubuntu Server 11.10 operating system. The Hadoop cluster which consists of these five nodes is organized as follows:

- One node is Namenode and Jobtracker
- For four remaining nodes, each node is the Tasktracker and Datanode

Each node is also running the Apache Mahout which is one of an Apache project that provides scalable machine learning algorithms.

5.2 Testing Data Set

The ten-percentage subset of the test set of KDD’99 is used for our experiments. The ten-percentage subset of the test set of KDD’99 is popular used for the evaluation of that of the proposed intrusion detection system. The prominent of KDD’99 data set is that each connection is labeled as either normal, or as an attack, with exactly one specific attack type. Moreover, additional attack types which not included in the training data set of KDD'99 are included in the testing data set of KDD'99. Therefore, these additional attack types make the data set more realistic. In addition, the additional attack types are the challenging tasks for the evaluative system. The details of the ten-percentage subset of the test set of KDD'99 are shown in the Table 5.1.

Table 5.1: The redundant records and ratio of the distinct records
of the ten-percentage subset of the test set of KDD'99

Labeled	Original Records	Distinct Records	Reduction Rate	Ratio of Distinct Records
Normal	60,593	47,913	20.93%	61.99%
Attacks	250,436	29,378	88.27%	38.01%
Total	311,029	77,291	75.15%	100.00%

According to the Table 5.1, the ten-percentage subset of the KDD'99 testing set has the redundant records approximately 75% of the entire records. After the reduction, the normal records are approximately 62% of the entire distinct records. The number of record of the data set after the reduction is approximately seventy seven thousand records.

5.3 Experimental Results

In this Section, the results of the experiments are discussed. The results of the first clustering using K-Means algorithm are shown in the Section 5.3.1. Section 5.3.2 shows the results of the first Association Rule Mining using Parallel FP-Growth and the Normal Profiles. After that, the second clustering using K-means is applied and their results are shown in Section 5.3.3. Finally, the characteristics of the suspect intrusions are described in Section 5.3.4.

5.3.1 The 1st Clustering using K-Means algorithm

After we remove the redundant records, K-Means algorithm is applied to create the normal profiles for this data set. According to the Table 4.15, the appropriate number of K with the approximately seventy seven thousand records of data set is 25. However, the data set is the ten-percentage subset of KDD’99 testing set. Therefore, Figure 5.1 and Figure 5.2 will show the accuracy rate with the various numbers of initial clusters (K) and the Silhouette index value with the various numbers of initial clusters (K) respectively.

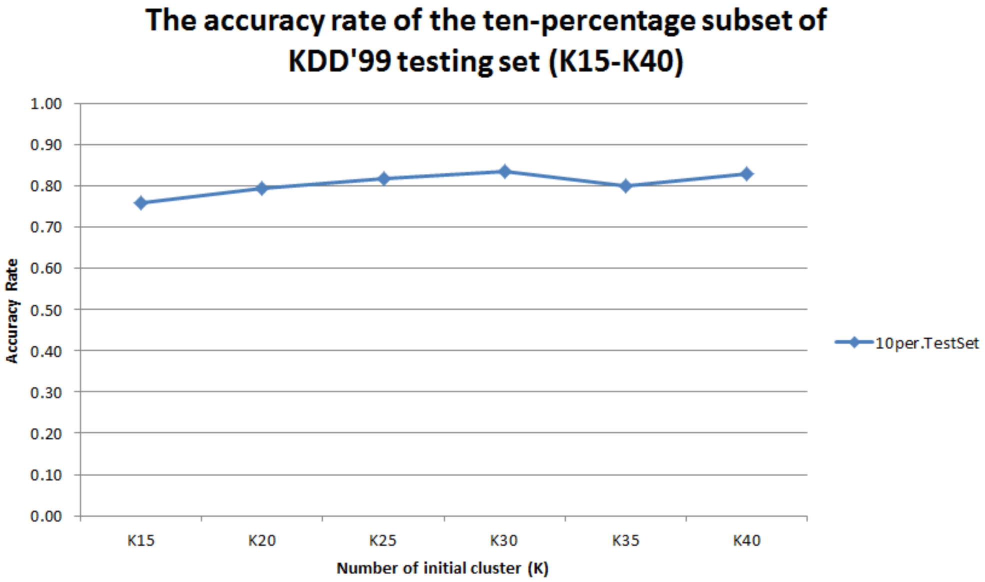


Figure 5.1: The accuracy graph of the ten-percentage subset of KDD’99 testing data set with number of initial clusters (K) ranging from 15 to 40 (period of 5)

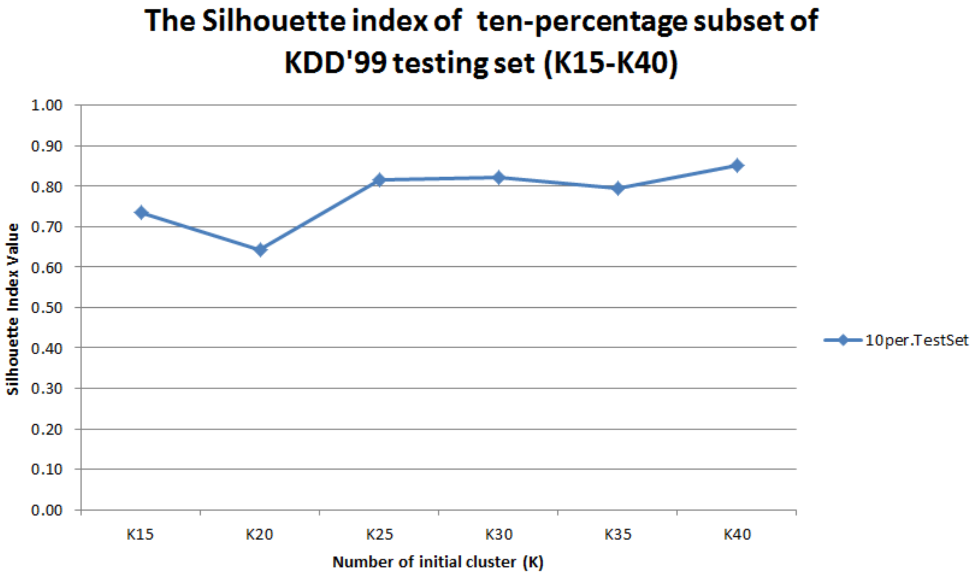


Figure 5.2: The Silhouette index graph of the ten-percentage subset of KDD’99 testing data set with number of initial clusters (K) ranging from 15 to 40 (period of 5)

According to the Figure 5.1and Figure 5.2, K25 and K30 are determined to be an appropriate number of K for the ten-percentage subset of KDD'99 testing set. These two numbers of K produce the value which is higher than 0.80 and their values are so close. Therefore, K25 is determined to be an appropriate number of K for the ten-percentage subset of KDD'99 testing set. For more information, the details of the results of the Silhouette index and the confusion matrix of the ten-percentage subset of KDD'99 testing set are shown in the Table 5.2 and Table 5.3 respectively.

Table 5.2: The details of the results of the Silhouette index of the ten-percentage subset of KDD'99 testing set

Number of K	Silhouette Index Value
K15	0.73521
K20	0.64318
K25	0.81611
K30	0.82109
K35	0.79536
K40	0.85024

Table 5.3: The details of the confusion matrix of the ten-percentage subset of KDD'99 testing set

Matrix	K15	K20	*K25*	K30	K35	K40
True Positive Rate (TPR)	0.94342	0.90674	*0.89672*	0.91966	0.92082	0.91966
False Positive Rate (FPR)	0.35458	0.27632	*0.23211*	0.21751	0.27507	0.22665
True Negative Rate (TNR)	0.64542	0.72368	*0.76789*	0.78249	0.72493	0.77335
False Negative Rate (FNR)	0.05658	0.09326	*0.10328*	0.08034	0.07918	0.08034
Detection Rate (PPV)	0.61981	0.66785	*0.70302*	0.72151	0.67226	0.71315
Accuracy	0.75864	0.79323	*0.81684*	0.83461	0.79936	0.82893

With the appropriate number of K in hand, the ten-percentage subset of KDD'99 testing set is partitioned by using the K-Means algorithm based on Apache Mahout. The maximum iteration for clustering is 10. Therefore, the results of the clustering are shown in the Table 5.4.

Table 5.4: The results of the first clustering (K25) of the ten-percentage subset of KDD'99 testing set

Confusion Matrix	K25
True Positive Rate (TPR)	0.89672
False Positive Rate (FPR)	0.23211
True Negative Rate (TNR)	0.76789
False Negative Rate (FNR)	0.10328
Positive Predictive Value (PPV)	0.70302
Negative Predictive Value (NPV)	0.92386
False Discovery Rate (FDR)	0.29698
Accuracy	0.81684

According to the Table 5.4, The TPR rate and accuracy rate is greater than 0.80. However, the FPR rate is slightly greater than 0.20 and the TNR rate and detection rate is slightly less than 0.80. The high rates of TPR with low rate of FPR means that the attack patterns are highly correctly identified as the intrusion and there is rarely misidentified the normal patterns as the intrusion. The high rate of TNR with low rate of FNR is meaning that the normal patterns are highly correctly identified as the normal activities and there is rarely misidentified the attack patterns as the normal activities.

The Positive Predictive Value (PPV) or the detection rate is the proportion of attack patterns which are correctly identified as the intrusion to all the predicted attack patterns. The detection rate in our experiments is approximately 0.70. Therefore, the high detection rate is meaning that the system is precisely detecting the intrusions.

The high Negative Predictive Value (NPV) [33] is meaning that the system is rarely misidentified attacks as the normal activities. In our experiments, the NPV rate is approximately 0.92 which is quite high NPV rate.

The False Discovery Rate (FDR) [34] is the expected percentage of false positive among the entire predictions. For example, if the system identifies 100 attack patterns as intrusions with the false discovery rate 0.3, there are 70 attack patterns which should be correctly identified as intrusions [35]. In our experiments, the FDR rate is approximately 0.29.

5.3.2 The 1st Association Rule Mining using Parallel FP-Growth

In this part, the normal profiles are created using the Parallel FP-Growth algorithm based on Apache Mahout. Each cluster is analyzed using Parallel FP-Growth algorithm to discover the frequently patterns which occur more than 70% of the minimum threshold. After that, the generated pattern rules which the numbers of attribute are in the range of 58% to 65% of the number of the entire attributes are determined to be the normal profiles of the system. A part of the generated normal profiles are shown in the Figure 5.3.

With the range of 58% to 65% of the number of the entire attributes, the pattern rules which have 24 to 26 attributes are selected to be the normal profiles. Therefore, there are 198 rules are created. The example rules of these 198 rules are shown in the Figure 5.3. For each rule, the attributes are separate by the space. For each attribute, the number of sequence of that attribute is a number in the parentheses (“[”, “]”). For example, the rule “[02]tcp [03]http [04]SF [07]0” consist of 4 attributes. They are the 2-nd attribute, the 3-rd attribute, the 4-th attribute and the7-th attribute with the value tcp, http, SF and 0 respectively.

After the normal profiles are generated, all of the records are compared to the normal profiles. If any records are matched to the normal profiles, that record is removed from the data set. Thus, the remaining records are the suspect records to be intrusions. The detailed results of the detection with the normal profiles are shown in the

Table 5.5. Moreover, the confusion matrix of the results of the detection with the normal profiles is also shown in the Table 5.6.

Table 5.5: The detailed results of the detection with the normal profiles

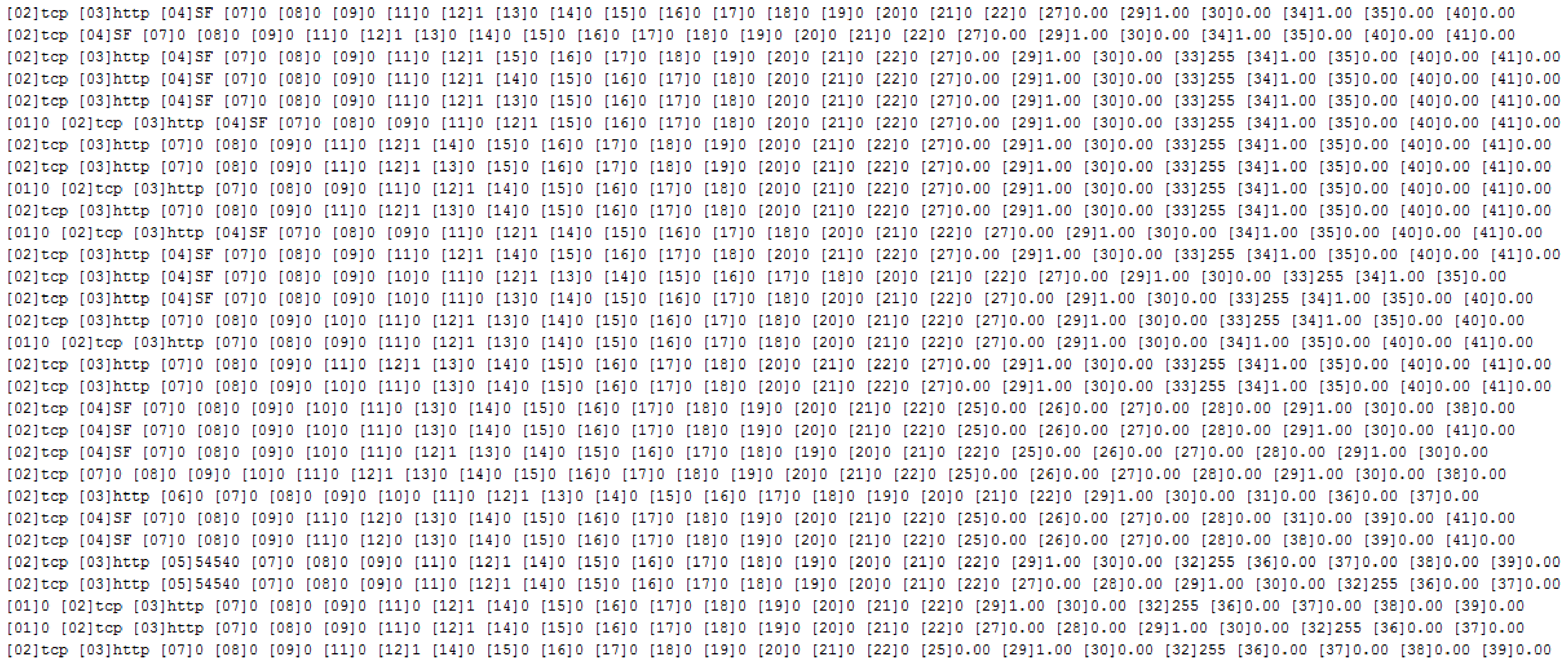
Labeled	Original Records	Removed Records	Remaining Records
Normal	47,913	47,227	686
Attacks	29,378	5,534	23,844
Total	77,291	52,761	24,530

Table 5.6: The confusion matrix of the results of the detection with the normal profiles

Confusion Matrix	Atrribute 58% - 65%
True Positive Rate (TPR)	0.81163
False Positive Rate (FPR)	0.01432
True Negative Rate (TNR)	0.98568
False Negative Rate (FNR)	0.18837
Positive Predictive Value (PPV)	0.97203
Negative Predictive Value (NPV)	0.89511
False Discovery Rate (FDR)	0.02797
Accuracy	0.91952

According to the

Table 5.5 and Table 5.6, the generated normal profiles produce the high rate of TPR, TNR, PPV, NPV and accuracy with the low rate of FPR, FNR and FDR. There are 47,227 normal records removed from the data set. However, 5,534 attack records are also removed from the data set. Nonetheless, the remaining records are almost the attack records. There are only 686 normal records among 24,530 records. Therefore, the generated normal profiles of the system are reliable and precise. The remaining records will be analyzed in the next step.



```
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [34]1.00 [35]0.00 [40]0.00
[02]tcp [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[01]0 [02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[01]0 [02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[01]0 [02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [10]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00
[02]tcp [03]http [04]SF [07]0 [08]0 [09]0 [10]0 [11]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [10]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00
[01]0 [02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [10]0 [11]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [20]0 [21]0 [22]0 [27]0.00 [29]1.00 [30]0.00 [33]255 [34]1.00 [35]0.00 [40]0.00 [41]0.00
[02]tcp [04]SF [07]0 [08]0 [09]0 [10]0 [11]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [27]0.00 [28]0.00 [29]1.00 [30]0.00 [38]0.00
[02]tcp [04]SF [07]0 [08]0 [09]0 [10]0 [11]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [27]0.00 [28]0.00 [29]1.00 [30]0.00 [41]0.00
[02]tcp [04]SF [07]0 [08]0 [09]0 [10]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [27]0.00 [28]0.00 [29]1.00 [30]0.00
[02]tcp [07]0 [08]0 [09]0 [10]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [27]0.00 [28]0.00 [29]1.00 [30]0.00 [38]0.00
[02]tcp [03]http [06]0 [07]0 [08]0 [09]0 [10]0 [11]0 [12]1 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [29]1.00 [30]0.00 [31]0.00 [36]0.00 [37]0.00
[02]tcp [04]SF [07]0 [08]0 [09]0 [11]0 [12]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [27]0.00 [28]0.00 [31]0.00 [39]0.00 [41]0.00
[02]tcp [04]SF [07]0 [08]0 [09]0 [11]0 [12]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [27]0.00 [28]0.00 [38]0.00 [39]0.00 [41]0.00
[02]tcp [03]http [05]54540 [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [29]1.00 [30]0.00 [32]255 [36]0.00 [37]0.00 [38]0.00 [39]0.00
[02]tcp [03]http [05]54540 [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [28]0.00 [29]1.00 [30]0.00 [32]255 [36]0.00 [37]0.00
[01]0 [02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [29]1.00 [30]0.00 [32]255 [36]0.00 [37]0.00 [38]0.00 [39]0.00
[01]0 [02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [28]0.00 [29]1.00 [30]0.00 [32]255 [36]0.00 [37]0.00
[02]tcp [03]http [07]0 [08]0 [09]0 [11]0 [12]1 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [29]1.00 [30]0.00 [32]255 [36]0.00 [37]0.00 [38]0.00 [39]0.00
```

Figure 5.3: The example of the generated normal profiles

5.3.3 The 2nd Clustering using K-Means algorithm

In this process, the remaining records will be partitioned using K-Means algorithm based on Apache Mahout once again. For this clustering, we apply K-Means clustering with 25 initial centroids and 10 maximum iterations. After the clustering, the largest cluster is labeled to be an intrusion. Also the clusters which are greater than one-fourth of the largest cluster are labeled to be intrusions. The summary of number of records of each cluster after the 2nd clustering is shown in Table 5.7.

Table 5.7: The summary of number of records of each cluster after the 2nd clustering

Number of Cluster	Summary	Number of Cluster	Summary
C1	94	C12	1
C2	78	C13	*13,626*
C3	70	C14	6,502
C4	141	C15	688
C5	707	C16	33
C6	59	C17	133
C7	144	C18	14
C8	23	C19	402
C9	48	C20	1,137
C10	60	C21	3
C11	52		

Summary	24,015	Max Cluster/4:	3,406.50

According to the Table 5.7, there are 21 clusters. The 13-th cluster is the largest cluster which has 13,626 records. Therefore, one-fourth of the largest cluster is 3,406.5 records. However, there are 25 initial centroids but the results of the clustering have only 21 clusters. This incident occurs when there are some initial centroids which do not have any member to be assigned to them. Thus, there are only 21 remaining clusters as the results of clustering. The details of each cluster and its label are shown in the Table 5.8.

Table 5.8: The details of each cluster and its label from the 2nd clustering

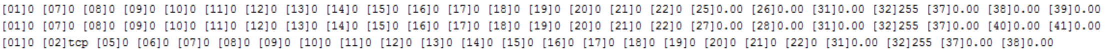
Number of Cluster	Normal	Attack	Sum	Labeled
C1	0	94	94	Normal
C2	39	39	78	Normal
C3	30	40	70	Normal
C4	1	140	141	Normal
C5	3	704	707	Normal
C6	40	19	59	Normal
C7	0	144	144	Normal
C8	13	10	23	Normal
C9	24	24	48	Normal
C10	9	51	60	Normal
C11	7	45	52	Normal
C12	1	0	1	Normal
C13	*0*	*13,626*	*13,626*	*Attack*
C14	*0*	*6,502*	*6,502*	*Attack*
C15	234	454	688	Normal
C16	30	3	33	Normal
C17	40	93	133	Normal
C18	3	11	14	Normal
C19	35	367	402	Normal
C20	112	1,025	1,137	Normal
C21	3	0	3	Normal
Sum:	624	23,391	24,015	
		Max/4:	3,406.5	

According to the Table 5.8, the 13-th cluster is the largest cluster and there is another cluster, the 13-th cluster, which has the number of records greater than one-fourth of the largest cluster. Therefore, the 13-th and the 14-th clusters are labeled to be intrusions. The 13-th and the 14-th clusters will be analyzed in the next process. However, there are some attack patterns in other clusters. The detected attack patterns and undetected attack patterns of our system are shown in the next process.

5.3.4 The 2nd Association Rule Mining using Parallel FP-Growth

In this part, the characteristics of the suspect intrusion are proposed using the Parallel FP-Growth algorithm based on Apache Mahout. According to the previous process, there are two suspect clusters. These clusters will be analyzed by the Parallel FP-Growth. Finally, the characteristics of these intrusions are created.

The minimum threshold of this process is 70%. After the analyzed, the rules which have the number of attribute in the range of 58% to 65% are determined to be the characteristics of these intrusions. The examples of the generated characteristic of these intrusions are shown in Figure 5.4.



```
[01]0 [07]0 [08]0 [09]0 [10]0 [11]0 [12]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [25]0.00 [26]0.00 [31]0.00 [32]255 [37]0.00 [38]0.00 [39]0.00
[01]0 [07]0 [08]0 [09]0 [10]0 [11]0 [12]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [27]0.00 [28]0.00 [31]0.00 [32]255 [37]0.00 [40]0.00 [41]0.00
[01]0 [02]tcp [05]0 [06]0 [07]0 [08]0 [09]0 [10]0 [11]0 [12]0 [13]0 [14]0 [15]0 [16]0 [17]0 [18]0 [19]0 [20]0 [21]0 [22]0 [31]0.00 [32]255 [37]0.00 [38]0.00
```

Figure 5.4: The examples of the generated characteristic of the suspect intrusions

According to the Figure 5.4, they are the examples of the total 124 characteristic rules. The details of the number of the generated rules and the number of the records in a data set are shown in Table 5.9. Moreover, the most of these characteristic are the Neptune and Satan attacks which are one of the DoS attack types. Therefore, DoS attacks are the eminent attack types for the intrusion detection with our system. However, there are three remaining attack types in KDD’99 data set. There are U2R (User to Root), R2L (Remote to Local) and Probing attacks which are not the eminent attack types for intrusion detection with our system.

Table 5.9: The details of the number of the generated rules

and the number of the records in a data set

Original Data Set (Records)	Generated Rules (Records)	Generated Rules/Original Data Set (Percentage)
311,029	124	0.04%

According to the Table 5.9, the number of the generated rules is 124 records from the original data set which has 311,029 records. These generated rules are 0.04% of all the number of the original data set. Therefore, these rules are very concise.

5.4 The Comparison

In this Section, the results of our system are compared to other system. Our system does not need any learning phases is one of the advantages of our system. Moreover, the dimension reduction is not required. The comparison of our system and other systems are shown are classified into 3 groups. (1) The systems which apply classification technique for intrusion detection, (2) The systems which apply clustering technique for intrusion detection and (3) The systems which apply classification and clustering technique (Hybrid) for intrusion detection.

5.4.1 The System using Classification Technique for Intrusion Detection

The classification technique is one of the data mining techniques. It consists of two main steps. There are (1) learning phase and (2) recognition phase. The classification technique is also called the supervised learning technique. In contrast, clustering technique is also called unsupervised learning technique. There are many algorithms of the classification techniques such as Support Vector Machine (SVM), Naïve Bayes and K-nearest neighbors.

In [32], a group of researchers applies classification techniques for the intrusion detection. They apply J48, Naïve Bayes, NB Tree, Random Forest, Random Tree, Multi-layer Perceptron and SVM with the KDD’99 data set. They also proposed a new data set, NSL-KDD, which is the KDD’99 data set that removed the duplicated records from the data set. The redundant records will bias the algorithm to the frequent records and prevent the algorithm from the infrequent records. Therefore, the duplicated records are removed. The results of their experiments [32] and our results are compared and shown in the Figure 5.5.

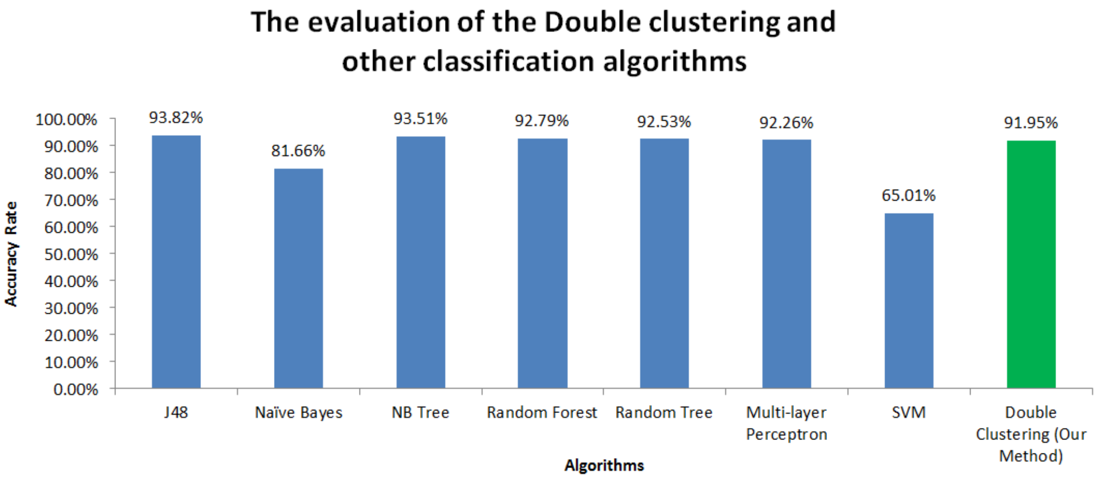


Figure 5.5: The results of the classification techniques [32] and the double clustering techniques of the KDD'99 data set (removed duplicated records)

According to the Figure 5.5, our method, the double clustering techniques, produces 91.95% of accuracy rate. The Other classification methods approximately produce 92%-93% of accuracy rate.

Another group of researchers proposed the anomaly based intrusion detection using meta-ensemble classifier [36]. They apply C4.5 algorithm, Naïve Bayes classifier and decision table as the ensemble model. The ensemble model [37] is a technique to create a strong learner from many of the weak learners. The results from many learners will be voted to create the final result. The results of their experiments and our results are compared and shown in Figure 5.6 [36].

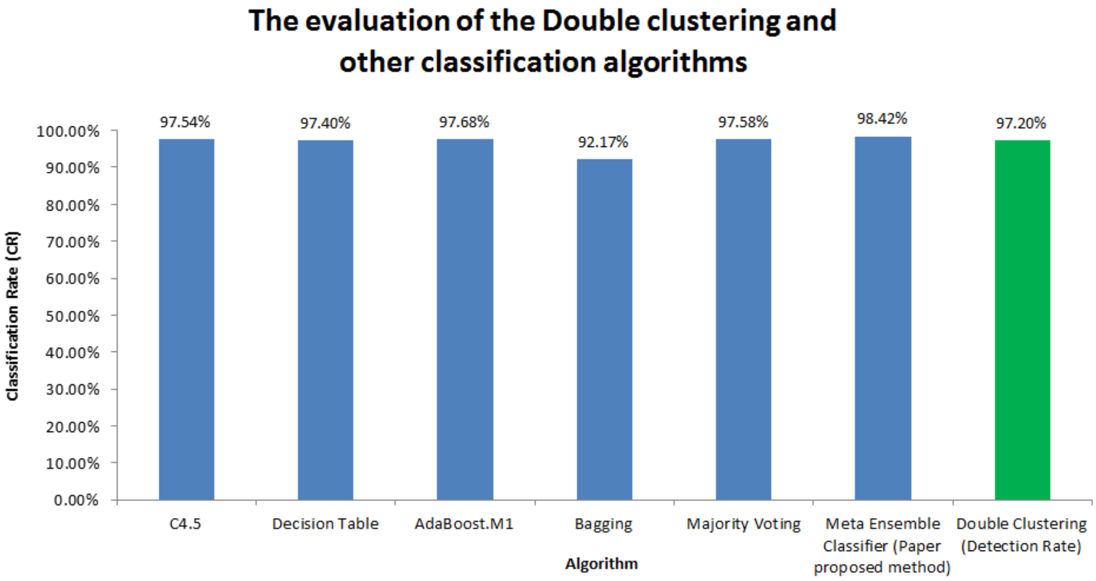


Figure 5.6: The results of the classification techniques, the Meta Ensemble technique [36] and the double clustering techniques of the KDD'99 data set

According to the Figure 5.6, the Meta Ensemble classifier produces 98.42% of classification rate. The other classification algorithms approximately produce 97% of classification rate. However, the Double clustering technique which is our proposed method also approximately produces 97.20% of detection rate.

The double clustering method which is our proposed method is based on unknown knowledge. It does not need any learning phases. In contrast, the others classification methods need the learning phases to recognition. Therefore, the classification methods should produce better results of accuracy rate with their learning data set. Nonetheless, the double clustering techniques which is our proposed method, and the others classification methods produce the similarly accuracy rate.

5.4.2 The System using Clustering Technique for Intrusion Detection

The clustering technique is one of the data mining techniques. The main concept of the clustering is partitioning data into groups. Each group consists of the same behaviors of data. Therefore, the data in the same group more resemble to the other data in its group than the data in other groups. However, the clustering method does not need any learning phases.

In [38], a group of researchers proposed an adaptive clustering for intrusion detection based on wavecluster algorithm which is used in image processing. They develop and apply the wavecluster algorithm for intrusion detection. The results of their experiments and our result are compared and shown in Figure 5.7.

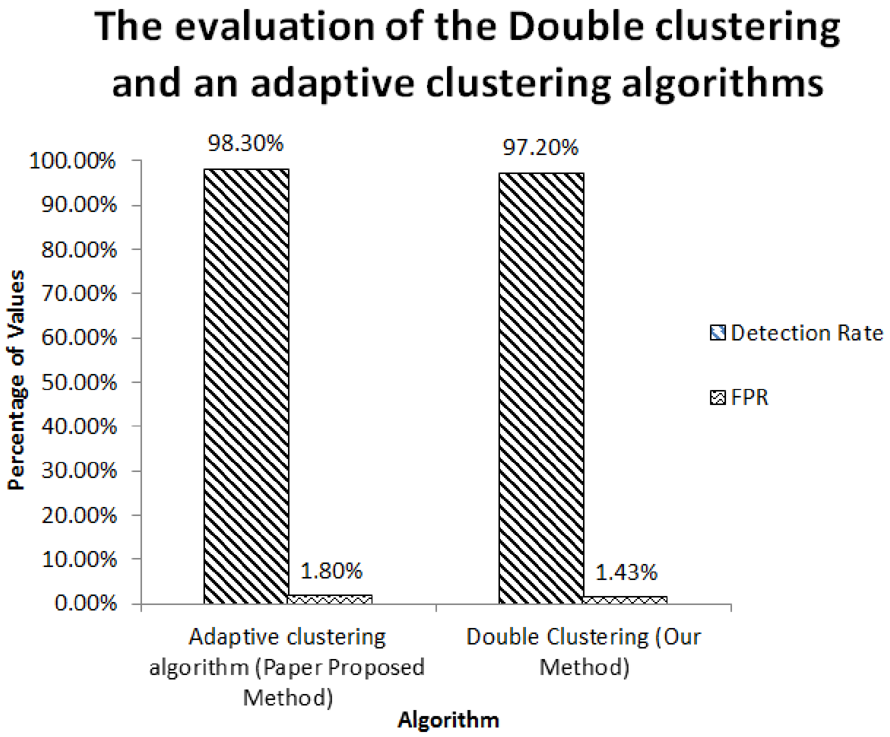


Figure 5.7: The results of an adaptive clustering algorithm [38] and the double clustering techniques of the KDD’99 data set

According to the Figure 5.7, the adaptive clustering with wave cluster algorithm produces 98.30% of detection rate with 1.80% of false positive rate. However, our method produces 97.20% of detection rate which is slightly less than that of the adaptive clustering. Nonetheless, our method also yields the false positive rate less than that of the adaptive clustering algorithm.

5.4.3 The System using Classification and Clustering Technique (Hybrid) for Intrusion Detection

The hybrid system for intrusion detection is another technique to enhance the efficiency of the intrusion detection. The hybrid technique combines the classification technique and clustering technique together.

In [39], a group of researchers proposed the hybrid intrusion detection system. They apply one clustering algorithm and two classification algorithms. They are K-means algorithm, K-Nearest Neighbors (KNN) algorithm and Naïve Bayes algorithm. Moreover, they reduce the number of attributes by using an entropy based feature selection algorithm. The KDD'99 data set is used for the evaluations and their results of the experiments and our result are compared and shown in Figure 5.8.

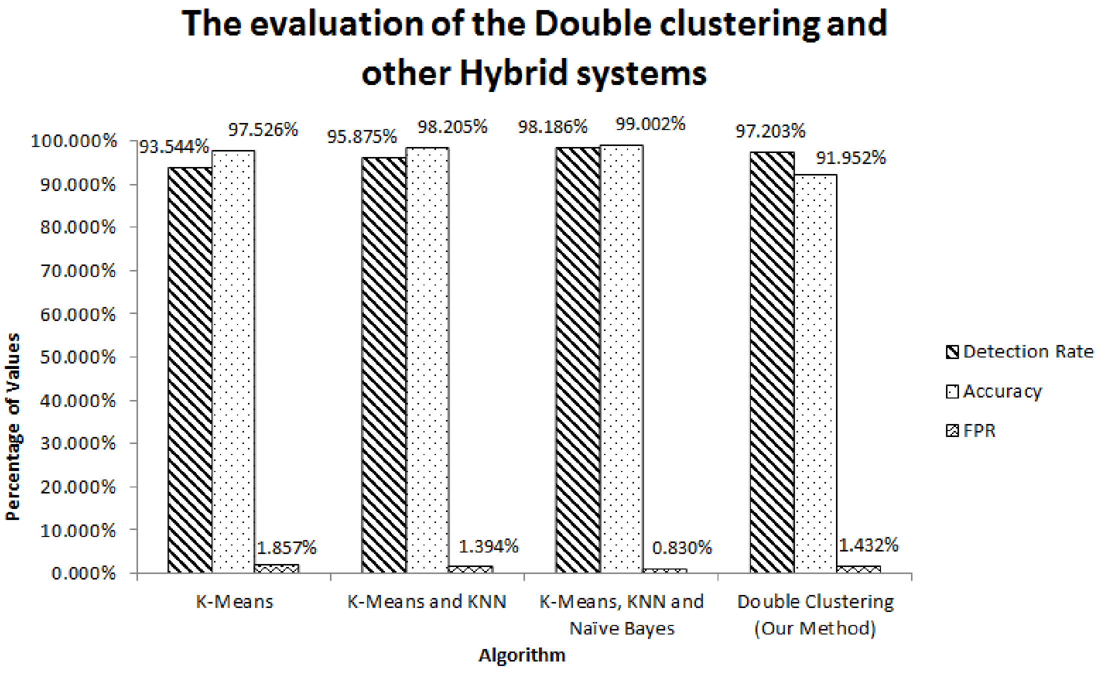


Figure 5.8: The results of the Hybrid system for intrusion detection [39] and the double clustering techniques of the KDD'99 data set

According to the Figure 5.8, the hybrid systems produce the detection rate and accuracy rate higher than the clustering algorithm. Their proposed method which is the combining of K-means, KNN and Naïve Bayes algorithm approximately produce 98% and 99% of detection rate and accuracy rate respectively. However, the double clustering which is our proposed method produces the detection rate as good as the hybrid system. Nonetheless, the accuracy rate of the double clustering is less than the hybrid system. The double clustering also produces the false positive rate greater than the hybrid system.

5.5 Discussion and Conclusion of the Experiments

In our experiments, the normal profiles and the characteristics of the suspect intrusions are generated. With our methods, we can apply the double clustering technique based on Apache Mahout for log analysis using the anomaly detection method. Our generated normal profiles are precise with the 97.20% of detection rate and 1.43% of false positive rate (false alarm rate) for the ten-percentage subset of KDD'99 testing set. Moreover, the generated characteristics are concise and useful to enhance the security of the system.

In our method, KDD'99 data set is used for the evaluation with the confusion matrix. The confusion matrix is commonly used with the KDD'99 data set in a lot of researches. There are many index values in confusion matrix which are compared to other researches. Therefore, the confusion matrix is used to evaluate our experiments. However, the KDD'99 data set has the label for each record. Thus, the confusion matrix can be applied with the KDD'99 data set.

For other data set which is the non-labeled data, the Silhouette index is applied to evaluate the system. From our experiments as shown in Figure 5.1 and Figure 5.2, they show that the results of Silhouette index are comparable to the results of confusion matrix. The results of the Silhouette index are similarly to the results of the accuracy rate of confusion matrix. Their trends of the Silhouette index value graph which are used to determine an appropriate number of initial clusters (K) for K-means algorithm are in the same way as the trends of the accuracy rate graph. Therefore, the Silhouette index is able to use as the evaluation index for other non-labeled data set.

Finally, the results of our method are the normal profiles and the characteristics of the suspect intrusions. The characteristics of the suspect intrusions show that the DoS attacks are the eminent attack types for the intrusion detection with our system.

CHAPTER VI
CONCLUSION

In this dissertation, we proposed the implementation of applying Mahout for log analysis to support firewall rules generation. The normal profiles and the characteristics of the suspect intrusions are presented. Our normal profiles produce the precise results with the TPR rate, TNR rate, accuracy rate and detection rate which are almost greater than 0.80. Moreover, our implementation can generate the characteristics of the suspect intrusions with the percentage of confidence greater than 70%. Furthermore, the system can discover the frequently patterns among the large data set. However, DoS attacks are the eminent attack types for intrusion detection with our system.

Our implementation is based on the Clustering and association rules mining techniques. Therefore, the prominent of our system is that the system does not need the training or the learning step. Moreover, the system is scalable with the Apache Hadoop framework which is a framework for the distributed system and cluster for processing large set of data using a MapReduce programming model.

6.1 Contributions

This thesis has made the following contributions:

1. We can apply Apache Hadoop and Apache Mahout Framework for the large scale log analysis.
2. The appropriate number of K and the proper number of iterations of the K-Means algorithm based on Apache Mahout are proposed with the various size of data set.
3. Our system can generate the normal profiles which have the TPR rates, TNR rates and accuracy rates are greater than 0.80 without the training or learning step.
4. Our system can generate the characteristics of the suspect intrusions with the percentage of confidence greater than 70%.

6.2 Future Works

In our system, KDD’99 data set is used for evaluation. Therefore, the other data set of log files are used for the evaluation in the future. Moreover, we will propose a method to distinguish the type of attacks and generate their characteristics. These characteristics are useful to enhance the security of the system. In additions, the firewall rules are generated.

References

[1] D. Denatious and A. John. Survey on data mining techniques to enhance intrusion detection. Computer Communication and Informatics (ICCCI), 2012 International Conference on, 2012.

[2] E. Saboori, S. Parsazad and Y. Sanatkhani. Automatic firewall rules generator for anomaly detection systems with Apriori algorithm. Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on, 2010.

[3] Apache Hadoop. [Online]. 2008. Available from: http://hadoop.apache.org. [2012. June 29].

[4] Overview of Mahout. [Online]. 2012. Available from: https://cwiki.apache.org/confluence/display/MAHOUT/Overview. [2012, June 29].

[5] Apache Mahout. [Online]. 2008. Available from: http://en.wikipedia.org/wiki/Apache_Mahout. [2012, June 29].

[6] MapReduce. [Online]. 2005. Available from: http://en.wikipedia.org/wiki/MapReduce. [2012, June 29].

[7] J. Therdphapiyanak and K. Piromsopa, Applying Hadoop for log analysis toward distributed IDS. Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication (ICUIMC '13), 2013.

[8] J. Therdphapiyanak and K. Piromsopa. An analysis of suitable parameters for efficiently applying K-Means clustering to large TCPdump data set using Hadoop framework. ECTI-CON 2013 Proceedings of 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, in press.

[9] Intrusion Detection System. [Online]. Available from: http://en.wikipedia.org/wiki/Intrusion_detection_system. [2012, June 29].

[10] F. Sabahi and A. Movaghar. Intrusion Detection: A Survey. Systems and Networks Communications, 2008. ICSNC '08. 3rd International Conference on, 2008.

[11] k-means clustering. [Online]. 2005. Available from: http://en.wikipedia.org/wiki/K-means_clustering. [2012, June 29.

[12] M. Halkidi, Y. Batistakis and M. Vazirgiannis. Clustering algorithms and validity measures. Scientific and Statistical Database Management, 2001. SSDBM 2001. Proceedings. Thirteenth International Conference on, 2001.

[13] P.-N. Tan, M. Steinbach and V. Kumar. Introduction to Data Mining. 1. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005.

[14] L. Zhou, Z. Zhong, J. Chang, J. Li, J. Huang and S. Feng. Balanced parallel FP-Growth with MapReduce. Information Computing and Telecommunications (YC-ICT), 2010 IEEE Youth Conference on, 2010.

[15] Data Mining Algorithms In R/Frequent Pattern Mining/The FP-Growth Algorithm. [Online]. 2010. Available from: http://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Frequent_Pattern_Mining/The_FP-Growth_Algorithm. [2012, August 22].

[16] Confusion matrix. [Online]. 2004. Available from: http://en.wikipedia.org/wiki/Confusion_matrix. [2012, November 15].

[17] Sensitivity and specificity. [Online]. 2006. Available from: http://en.wikipedia.org/wiki/Sensitivity_and_specificity. [2012, November 15].

[18] Receiver operating characteristic. [Online]. 2003. Available from: http://en.wikipedia.org/wiki/Receiver_operating_characteristic. [2012, November 15].

[19] T. Fawcett. An introduction to ROC analysis. Pattern Recogn. Lett., 27 (June 2006) : 861-874.

[20] S. Petrović. A Comparison Between the Silhouette Index and the Davies-Bouldin Index in Labelling IDS Clusters. Proceedings of the 11th Nordic Workshop on Secure IT-systems (NORDSEC), 2006.

[21] P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20 (1987) : 53 - 65.

[22] Silhouette (clustering). [Online]. 2009. Available from: http://en.wikipedia.org/wiki/Silhouette_(clustering). [2012, November 12].

[23] Davies–Bouldin index. [Online]. 2009. Available from: http://en.wikipedia.org/wiki/Davies-Bouldin_index. [2012, November 12].

[24] C. Zhang, G. Zhang and S. Sun. A Mixed Unsupervised Clustering-Based Intrusion Detection Model. Genetic and Evolutionary Computing, 2009. WGEC '09. 3rd International Conference on, 2009.

[25] Y.-F. Zhang, Z.-Y. Xiong and X.-Q. Wang. Distributed intrusion detection based on clustering. Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on, 2005.

[26] W. Ren, J. Cao and X. Wu. Application of Network Intrusion Detection Based on Fuzzy C-Means Clustering Algorithm. Intelligent Information Technology Application (IITA), 2009.

[27] R. Naidu and P. Avadhani. A comparison of data mining techniques for intrusion detection. Advanced Communication Control and Computing Technologies (ICACCCT), 2012 IEEE International Conference on, 2012.

[28] A. Chandrasekhar and K. Raghuveer. Intrusion detection technique by using k-means, fuzzy neural network and SVM classifiers. Computer Communication and Informatics (ICCCI), 2013 International Conference on, 2013.

[29] Y. Liu, W. Pan, N. Cao and G. Qiao. System anomaly detection in distributed systems through MapReduce-Based log analysis. Advanced Computer Theory and Engineering (ICACTE), 3rd International Conference on, 2010.

[30] X. Y. Yang, Z. Liu and Y. Fu. MapReduce as a programming model for association rules algorithm on Hadoop. Information Sciences and Interaction Sciences (ICIS), 2010 3rd International Conference on, 2010.

[31] KDD bringing together the data mining, data science and analytics community. [Online]. 1997. Available from: http://www.sigkdd.org/kddcup/index.php?section=1999&method=info. [2012, September 20].

[32] M. Tavallaee, E. Bagheri, W. Lu and A. Ghorbani. A detailed analysis of the KDD CUP 99 data set. Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on, 2009.

[33] Negative predictive value. [Online]. 2005. Available from http://en.wikipedia.org/wiki/Negative_predictive_value. [2013, April 10].

[34] False discovery rate. [Online]. 2006. Available from: http://en.wikipedia.org/wiki/False_discovery_rate. [2013, April 10].

[35] The False Discovery Rate. [Online]. 2010. Available from: http://www.cbil.upenn.edu/PaGE/fdr.html. [2013, April 10].

[36] D. Boro, B. Nongpoh and D. K. Bhattacharyya. Anomaly based intrusion detection using meta ensemble classifier. In Proceedings of the Fifth International Conference on Security of Information and Networks (SIN '12), 2012.

[37] Ensemble learning. [Online]. 2009. Available from: https://en.wikipedia.org/wiki/Ensemble_learning. [2013, May 10].

[38] G. Wu, L. Yao and K. Yao. An Adaptive Clustering Algorithm for Intrusion Detection. Information Acquisition, 2006 IEEE International Conference on, 2006.

[39] H. Om and A. Kundu. A hybrid system for reducing the false alarm rate of anomaly intrusion detection system. Recent Advances in Information Technology (RAIT), 2012 1st International Conference on, 2012.

Biography

Jakrarin Therdphapiyanak was born in Bangkok, Thailand, on February, 1990. He received Bachelor of Computer Engineering from Chulalongkorn University, Thailand, on 2012. He is currently enrolling in Master of Computer Engineering Program at Chulalongkorn University, Thailand, on 2012.