Optimized Sampling with Clustering Approach for Large Intrusion Detection Data

Nani Yasmin[1], Anto Satriyo Nugroho[2], Harya Widiputra[3]
[1]*Faculty of Information Technology, Swiss German University*
German Centre Building 6F, Bumi Serpong Damai, 15321, Indonesia
[2]*Center for the Assessment & Application of Technology (PTIK-BPPT)*
Jalan MH Thamrin 8 BPPT 2nd bld. 4F, Jakarta, 10340, Indonesia
[3]*Knowledge Engineering and Discovery Research Institute, Auckland University of Technology*
7th Floor 350 Queen Street, Auckland, 1010, New Zealand
nani.yasmin@gmail.com, asnugroho@inn.bppt.go.id, harya.widiputra@aut.ac.nz

***Abstract*—Data mining is a process of discovering useful information from a data set. In data mining, there is a classification technique that depends on sampling accuracy to acquire a more accurate result in data classification or prediction. Therefore, a necessity in getting a good-quality sampling is required. The primary purpose of this research paper is to obtain the optimum sampling representing the original data set. Through sampling, we could minimize the total data that need to be processed. Because large amount of data requires longer processing time, reducing the amount of data with sampling will speed up the process of computing. In this study we introduced a new sampling algorithm with clustering approach applied to a network security data set. Preliminary results showed that proposed method offer fine result for large data set sampling.**

***Index Terms*—sampling algorithm, clustering, network security data set.**

I. INTRODUCTION

Nowadays, data mining is adequately popular in the information sciene area. The data culitivation is purposed to get the knowledge of data. Data mining is normally (but not limited to) dealing with large datasets and commonly employed in marketing, intrusion detection, surveillance, etc.For data that has many records, we may well use sampling in the data preprocessing step to lower the amount of data to be processed.

Sampling is a process of colleting several data which should represent the original data [1]. As the beginning of data exploration, sampling is also intended to gain maximum knowledge from the whole data set. Moreover, sampling also has immense influence to the final result of data analysis.

In recent years, a number of sampling algorithms have been introduced and proposed. The basic one is simple random sampling algorithms. According to Mc Call [2] "a **simple random** sample is one in which all elements of the population have an equal probability of being selected". Certainly, the sample will be selected in a random order. Sometimes simple random sample works acceptably for a certain types of data. Nevertheless, the problem of simple random sampling is when facing a large data set, it is often difficult or impossible to identify every element of the data, therefore extracted samples is usually not representing whole characteristics of the entire data set.

Another sampling algorithm is stratified random sampling [3]. In this method, data will be divided in to several strata such as ages, genders, and etc. The sample will be drawn in random order within each stratum, and each element must be assigned only into one group. In taking the random sample units, number of drawn samples has to be proportional to the size of the partition.Stratified sampling performs better than simple random sampling, since the sample will be represented all existing cases of the complete data set. On the other hand, this method would not be effective in non-homogeneous groups.

Currently, it is still difficult to find a proper way in getting a better sample from a large data set. Time consuming is a common problem that often shows up when we want to process a large data set. Yet, sampling is still required to diminish the processing time. Additionaly, accurate samples would influence final result of the data analysis. Therefore, it is critically important to have samples which represent the characteristics of the entire data set.

The aim of the study is to introduce a new sampling algorithm which obtains samples not in a random manner, but by considering similarity between data as well. The proposed algorithm will have capability in grouping data with similar characteristic and acquire the most representative data sample from every data collection as an output.

II. DESIGN AND IMPLEMENTATION

Our proposed method is an optimized sampling based on a clustering approach. Clustering approach is a technique to group data into some collections by taking

into account similarity of data described by its features. There are some examples of clustering algorithms which are known, i.e. K-Means clustering and ECM (Evolving Clustering Method). These methods are described in section 2.1 and 2.2 since it will be the foundation of the clustering process in our algorithm which will be elucidated in section 2.3

A. K-Means Clustering

K-Means clustering was introduced by J. McQueen [4]. K-Means is identified as one of the simplest and quietly efficient clustering technique. The prime principal of K-means is to define number of clusters to be created, *k* that appropriate for the data set. K-Means defined a model in terms of centroid, which is usually the mean of a group of points. The K-Means clustering algorithm works in three basic steps, as follow:

- **Step 1**: Determine the c or centroid coordinate .
- **Step 2**: Determine the distance of each object to the centroids, usually by using Euclidean distance.
- **Step 3**: Group the object by searching the minimum distance of centroid and the object.

First step of the algorithm is to decide how many clusters should be created to classify the data set. Next, the location of the centroid will be placed in randomly and certainly, the number of centroid will be matched (suited) with the sum of cluster that has been defined before. Then, distance between the centroid with each object is calculated, and the minimum distance will be determined. The new object then will be assigned to the centroid with the minimum distance. Basic approach to find distance between an object and each centroids is by implementing the Euclidean distance. After which, recalculate again distance between the centroid and all members inside the group and update the centroid position. Do the iteration until the object does not move group anymore.

Although K-Means clustering is quietly very simple and efficient, it also has some weaknesses in performing clustering which are; (1) the number of *k* must be determined by the user beforehand so the total cluster will depend on the user, and (2) it cannot handle data with different size.

B. Evolving Clustering Method - ECM

ECM (Evolving Clustering Method) is proposed by Song [5]. ECM is a dynamic clustering algorithm. An evolved node in an on-line form could represent a cluster centre of a distance based clustering method. Many clustering algorithms cannot update the cluster centre when a new data arrives. On contrary, ECM is able to renew the cluster centre and can as well create new clusters when new data come. It has been proven that ECM performs better compare to the other classic clustering methods (i.e. K-Means) and it can adapt to the changes of characteristic on the new data [5]. Steps of ECM algorithm are explained as follow:

- **Step 1**: Data point that comes first will be assigned as the first cluster with radius equal to zero.
- **Step 2**: The algorithm will be end when all data have presented. In case there is still incoming data, calculate the distance between the new point and all cluster centres.
- **Step 3**: If the distance between new point and a cluster centre is less than or equal with the cluster radius, the new point will be included into that cluster without changes the cluster centre or cluster radius. Then go to step 4
- **Step 4**: Calculate s distance which is the distance from new point to radius from each cluster. Find the minimum distance from every cluster (*Smin*).
- **Step 5**: If Smin > 2*threshold, the new point will be not entered to any cluster. Then, the new point will make a new cluster with radius = 0. Then go to step 2.
- **Step 6**: If Smin <= 2* threshold, the new point will be entered to that cluster, so the cluster centre and the radius will be updated. The new radius = *Smin*/2, and the new cluster will be located between the new data and the old cluster center where distance of new cluster center to new data = new radius.

ECM uses the Euclidean distance to calculate distance. ECM is a dynamic clustering method and has capability to apply a good optimization. Moreover, some clustering and classification problems can be solved by using ECM [5]. Hence, in our proposed algorithm we employ ECM as the core clustering algorithm.

C. Optimized Sampling Algorithm with ECM

In this part, the sampling algorithm that we propose based on clustering approach is described. The clustering method will be employed to a group of data with similar characteristic. Consequently, each cluster will have different interest to the other clusters. Therefore, the sampling procedures will be performed in each cluster. It aimed to acquire representative sampling as the sampling tasks has covered all data with different characteristics in which of those has been grouped into clusters. The sampling algorithms which is proposed in this study, is described as follow:

- **Step 1**: Partition the data into various groups based on the class label of each data records. Data partitioning can be done based on, for example, gender, age, etc. In this study, data will be partitioned based on intrusion types in KDD CUP 1999 data. There are 23 types of intrusion which are defined as class labels in the data set as listed in table 1.
- **Step 2**: Apply clustering process with ECM to each partition created in Step 1.
- **Step 3**: After clusters have been created as result of

the clustering process, sample is drawn from each cluster in every partition.A number of data which have the shortest distance to the centroid will be chosen as samples. These data are expected to represent the corresponding cluster.

- **Step 4**: Final step of the algorithm is to combine entire samples which have been drawn from each cluster in every partition.

Table 1. Variety intrusions in KDD CUP 1999 data

No	Label	total_conn
1	back	2,203
2	buffer_overflow	30
3	ftp_write	8
4	guess_passwd	53
5	imap	12
6	ipsweep	12,481
7	land	21
8	loadmodule	9
9	multihop	7
10	neptune	1,072,017
11	nmap	2316
12	normal	972,780
13	perl	3
14	phf	4
15	pod	264
16	portsweep	10,413
17	rootkit	10
18	satan	15,892
19	smurf	2,807,886
20	spy	2
21	teardrop	979
22	warezclient	1020
23	warezmaster	20

Figure 1 illustrates how the complete data set is partitioned and how samples are drawn from each cluster centre.

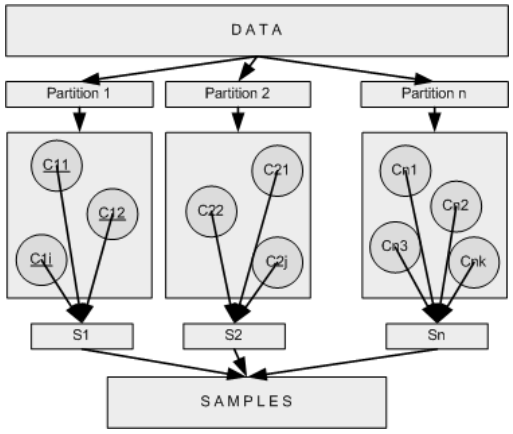

Fig 1. Illustration of how data set is partitioned, and how cluster centres are use as sample.

Validity of sampling results will be tested with statistical approach. The accuracy of the sampling will be measured from the data distribution of the samples compare to data distribution of the complete data set.

To calculate data distribution value, we employ standard deviation calculation as follow:

$$s = \left(\left(\frac{1}{n-1} \right) \sum_{i=1}^{n} (x_i - \bar{x}) \right) \quad (1)$$

where $$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \quad (2)$$

and n is number of elements in the sample. The sample is indicated to be closer with the mean when the standard deviation is low. If the standard deviation from the sample is closest with the entire data, we can assume that samples are representing the original data set.

III. Experiment and Discussion

In the experiment conducted to test the algorithm we have used KDD CUP 1999 data set [6]. KDD CUP 1999 data set is an intrusion data set which is used for The Third International Knowledge Discovery and Data Mining Tools Competition.This data is collected from raw TCP dump data. 42 attributes is mentioned in KDD CUP 1999. The data has a huge number of records which is around 4 million records. We aim to obtain samples with a size of 10-15% from the actual number of records. We believe with approximately 400,000 data samples, we could cover all data characteristics existing in the 4 million records

The KDD CUP 1999 data are not only formed in numerical ways, but also in form of combination of numeric and non numeric data (categorical data set). This categorical data should be labeled with a numeric values (regarding to distance calculation process in ECM). Therefore we labeled the categorical data with 1,2,3,4 and so on, for every object in categorical data. Another way to label categorical data is using 1-of-c coding [7] which we intend to implement in our future work.

As a first experiment, we took a sub-space of the KDD CUP 1999 data set as our real data and applied our proposed sampling algorithm on it. The first data we took was 729 data records. During the experiment data size will be increased step by step. Complete KDD CUP 1999 data set was not appropriate to be used because of resource and time limitation in conducting the experiment. This data as we have mentioned before, will be partitioned based on the intrusion type.

Some clustering algorithm works better with less but more significant attributes, therefore in this study we reduced number of attributes in the data set using a feature selection algorithm [8]. The main reason to this is

that we would like eliminate features that have less information in describing the data. The method of feature selection that we used is supervised filtered attributes with best first method. As the output of the feature selection algorithm, we got 9 attributes as the most informative attributes out of 42 attributes. Nevertheless, please note that there is a possibility the feature selection algorithm would give different output if the number of data is changed.

Our proposed algorithm uses ECM as its clustering method therefore it is required to define the minimum distance before the clustering process is started. In this experiment, we used 0.05 as a distance threshold. As a result the algorithm extracted 249 clusters from 729 data records which are divided into 23 partitions. Each cluster centre then taken as sample, thus number of drawn samples is approximately 34.15%.

Table 2. The results of attribute selection

No	Attributes name	Description
1	service	network service on the destination, e.g., http, telnet, etc.
2	flag	normal or error status of the connection
3	src_bytes	number of data bytes from source to destination
4	dst_bytes	number of data bytes from destination to source
5	land	1 if connection is from/to the same host/port; 0 otherwise
6	wrong_fragment	number of wrong fragments
7	num_root	number of root accesses
8	countx	number of connections to the same host as the current connection in the past two seconds
9	srv_count	number of connections to the same service as the current connection in the past two seconds
10	diff_srv_rate	% of connections to different services
11	dst_host_ srv_count	number of connections from the same host with same service to the destination host during a specified time window
12	dst_host_same_ src_port_rate	% of connections to same service ports from a destination host
13	dst_host_srv_ diff_host_rate	% of connections to the same service from different hosts to a destination host
14	dst_host_ rerror_rate	% of connections that have REJ errors from a destination host
15	label	attack types

note: to get complete description of KDD CUP 1990 data set attributes please refer to [9].

Next step we increased number of data to be sampled to 5,000 records. We re-applied the feature selection algorithm again and 15 attributes were chosen out of 42 attribute (shown in Table 2). In this second trial, we intended to have number of samples to be around 10.0-15.0% of the total records. We finally achieved 249 clusters from 5,000 records by setting the distance threshold value to 0.04. This gave us a number of samples approximately 9.2% of the whole data set. Additionally, we changed the distance threshold to 0.03 and received 620 clusters, which is around 12.4% out of the complete data set. Complete result of the experiment is shown in Table 3.

On the other hand, after the data is clustered in each partition, we found that there are several data that have different characteristic in every partition such as “smurf” since the standard deviation is still high (shown in Table 3). It shows that ECM can handle data with different interest and various characteristics are represented in the clusters formed by ECM.

Table 3. Cluster result in each partition, dthr = 0.03

No	Intrusion Type	Cluster	Std Dev.
1	back	36	4.5768
2	buffer_overflow	20	0.3999
3	ftp_write	7	2.5520
4	guess_passwd	18	11.0370
5	imap	12	5.1157
6	ipsweep	34	0.4318
7	land	12	1.5332
8	loadmodule	8	2.5453
9	multihop	7	2.5519
10	neptune	99	0.3190
11	nmap	38	0.3503
12	normal	113	0.3278
13	perl	3	11.1721
14	phf	3	2,116.8165
15	pod	33	2.5382
16	portsweep	34	0.3311
17	rootkit	9	2.5643
18	satan	43	0.3618
19	smurf	5	65.7723
20	spy	2	11.3022
21	teardrop	34	10.9822
22	warezclient	40	2.5539
23	warezmaster	10	2.5544
	Total Cluster	620	

Nonetheless, we recognize that the centroids of each cluster created by ECM are not a real data. The cluster center in ECM clustering is the result of mean calculations from all data samples which are belong to the cluster. Therefore, we believe that if we are able to find data samples in a cluster which have the shortest distance to the centroid, and use these data as drawn samples, we would be able to increase the sampling accuracy. In calculating the distance between data samples and their

centroid the Euclidean distance as describe in equation 3 can be employed.

$$d(x,y) = \sqrt{\sum_{k=1}^{n}(x_k - y_k)^2} \quad (3)$$

IV. Conclusion and Future Works

We have proposed a new algorithm that shows the ability to take a sample not in a random order, but by considering similar interest infatuated by data in the identical group. In this initial version of our proposed algorithm, samples are taken from the centroids of the clusters, which are the mean calculation of all data belong to the cluster. Yet, experiment results showed that the standard deviation of drawn samples can be considered to match the standard deviation of the complete data set in an acceptable degree, therefore we can conclude that the algorithm performs quite well for the KDD CUP 1999 data set.

As for future work, we would like to extend the algorithm to draw samples which are not the centroid of the cluster, but by taking into account those data in the cluster which are the closest ones to the centroid. Furthermore, we would like to investigate the possibility of using different method in labeling categorical data in order to observe if it would have any effect on the sampling results.

References

[1] P. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining. Boston: Pearson Education, Inc, 2006, pp. 47-50.

[2] R. McCall, Fundamental Statistics for psychology. New York: Harcourt Brace Jovanovich, 1980.

[3] T. Williams, L. Nozick, M. Samsalone, and R. Poston, "Sampling Techniques for Evaluating Large Concrete Structures: Part I," ACI Structural Journal., 2006, Vol. 103, No. 3, pp. 339-408.

[4] J. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability., Berkeley: University of California Press, 1967, 1:281-297.

[5] Q. Song, and N. Kasabov, "ECM - A novel on-line, evolving clustering method and its applications," In Proceedings of the fifth biannual conference on artificial neural networks and expert systems., New Zealand: Dunedin, 2001, pp. 87-92

[6] University of California, KDD CUP 1999 data, 1999, [Online]. Available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html[Accessed: 10 May 2009].

[7] C. Bishop, Neural Networks for Pattern Recognition. United States: Oxford University Press Inc., 1995

[8] I. Guyon, and A. Elissef, "An introduction to variable and feature selection," The Journal of Machine Learning Research., 2003, Vol. 3, pp. 1157-1182.

[9] S. Mukkamala, S. Andrew and A. Abraham, "Cyber Security Challenges: Designing Efficient Intrusion Detection Systems and Antivirus Tools," In Vemuri, V. Rao, Enhancing Computer Security with Smart Technology., 2006, USA: CRC Press, ISBN 0-8493-3045-9, pp. 125-163.