Intrusion detection system model using Fuzzy Kernel C-Means and Laplacian Score feature selection

Z Rustam and J Maharani
Department of Mathematics, Faculty of Mathematics and Natural Sciences (FMIPA), Universitas Indonesia, Kampus UI Depok, Depok 16424, Indonesia

Corresponding author’s e-mail: rustam@ui.ac.id

Abstract. Nowadays, where the technology dominates every field and activity like transaction, learning activities, private or corporate data storage, and so on are things that we have to look in detail especially in terms of security both in data storage and technology utilization. The trust of the security itself can be represented by the accuracy on model we used but the question is “Which model is good for the security?”, that question can be answered by the research or trial of models either by combining the models or construct the new one. Therefore, in this paper we tried an intrusion detection system model by combining Fuzzy Kernel C-Means method as a classifier and Laplacian Score method as a feature selection applied on KDD Cup 99 Data Set. As a result, we will see the comparison of the accuracy of intrusion detection system (IDS) model using Fuzzy Kernel C-means with and without Laplacian Score Feature Selection for the same data and features that resulted 60% and 67.06%, respectively.

1. Introduction
As we know, every activity that we do now is very easy to do. We are easy to do the communication, payment, data search, data storage, and so on. One of causes make our activities easy to do is the internet, so that internet has huge enthusiast and also nowadays researcher, student, lecturer, and maybe kids as an active user of internet.

Talk about the internet, network and computer science surely in charge on it. However, as the time goes by the increased of internet user, not a few people who misuse the internet as a place to do crime or usually we called cybercrime for examples stealing the money, stealing the identity, hacking, plagiarism, and so forth. Therefore, we think intrusion detection system (IDS) will be useful to handle or prevent this kind of problem and also IDS very popular among the researchers caused of the importance of intrusion detection on network.

In this paper, we proposed the idea for IDS model using classifier combined with future selection technique. We use Fuzzy Kernel C-Means (FKCM) as a classifier and Laplacian Scores Feature Selection (LSFS) as a feature selection.

On the chart, that is the step that actually we would like to do. We will compare the accuracy result with and without feature selection process.

2. Materials and methods

2.1. Data set
In this research, we use the KDDCup 99 Data Set. This data usually used for intrusion detection research. In this paper, we proposed the IDS model using feature selection and classifier to build a predictive model capable of distinguishing between "bad" connections, called intrusions or attacks, and "good" normal connection [1] as shown in figure 1.
Table 1. List of features on KDDCup 99 Data Set

| Features                  |          |
|---------------------------|----------|
| 1. duration               | 2. is_host_login                  |
| 3. protocol_              | 4. is_guest_login                  |
| 5. service                | 6. count                          |
| 7. flag                   | 8. srv_count                       |
| 9. src_bytes              | 10. serror_rate                    |
| 11. dst_bytes             | 12. srv_serror_rate                 |
| 13. land                  | 14. rerror_rate                    |
| 15. wrong_fragment        | 16. srv_rerror_rate                 |
| 17. urgent                | 18. same_srv_rate                   |
| 19. hot                   | 20. diff_srv_rate                   |
| 21. num_failed_logins     | 22. srv_diff_host_rate             |
| 23. logged_in             | 24. dst_host_count                  |
| 25. num_compromised       | 26. dst_host_srv_count              |
| 27. root_shell            | 28. dst_host_same_srv_rate         |
| 29. su_attempted          | 30. dst_host_diff_srv_rate         |
| 31. num_root              | 32. dst_host_diff_srv_rate         |
| 33. nu_file_creations     | 34. dst_host_same_src_port_rate    |
| 35. num_shells            | 36. dst_host_srv_diff_host_rate    |
| 37. num_access_file       | 38. dst_host_srv_serror_rate       |
| 39. num_outbond_cmds      | 40. dst_host_rerror_rate           |
| 41. is_host_login         | 42. dst_host_srv_rerror_rate       |

In this data set, there are 4 main categories of attacks: DoS (Denial of Service), R2L (Root to Local), U2R (User to Root), and Probing; and the other one as a Normal. On the table 1, there are 42 features on KDDCup 99 to select by feature selection to be a sub feature for an optimal data set.

2.2. Feature selection

Feature Selection is an important pre-processing step in machine learning and data mining due to the rapid accumulation of high-dimensional data [2]. There are 2 types of feature selection methods: wrapper method and filter method. The difference of wrapper method and filter method is based how they work. The filter method processes the data based on class labels (supervised) and the work of it is correlating between the features and class labels. The filter methods evaluate the goodness of features by using the intrinsic characteristics of the training data and are independent on any learning algorithm [2]. On the other contrary, wrapper method processes the data that has no class labels (unsupervised). the wrapper methods directly use predetermined learning algorithms to evaluate the features [3].
Besides of talking about feature selection types, feature selection has many functions of itself. Feature selection useful for reducing dimensionality, and removing irrelevant and redundant features that can make our data accurately classified.

2.2.1. Laplacian Score feature selection. Laplacian Scores is one of feature selection methods. It can work both in supervised and unsupervised. In this paper, we used Laplacian Score as a Feature Selection and with KDDCup 99 data set, we used supervised as a method. Laplacian Score analyses 2 points of data whether they’re close to each other or not. If they’re close to each other, probably they’re on the same class and the opposite for the other. Laplacian score evaluates the features by its locality preserving power of each feature. Laplacian score analyses based on local geometric structure, so we build a nearest neighbor graph and to this, laplacian score evaluates the features.

Laplacian Score algorithm. Let \( L_r \) denote the Laplacian Score of the \( r \)-th feature. Let \( f_{ri} \) denote the \( i \)-th sample of the \( r \)-th feature, \( i = 1,2,\ldots,m \). Our algorithm can be stated as follows [4]:

- Construct a nearest neighbor graph \( G \) with \( m \) nodes. The \( i \)-th node corresponds to \( x_i \). We put an edge between nodes \( i \) and \( j \) if \( x_i \) and \( x_j \) are “close”, i.e. \( x_i \) is among \( k \) nearest neighbors of \( x_j \) or \( x_j \) is among \( k \) nearest neighbors of \( x_i \). When the label information is available, one can put an edge between two nodes sharing the same label.
- If nodes \( I \) and \( J \) are connected, put \( S_{ij} = e^{-\frac{\|x_i-x_j\|^2}{t}} \), where \( t \) is a suitable constant. Otherwise, put \( S_{ij} = 0 \). The weight matrix \( S \) of the graph models the local structure of the data space.
- For the \( r \)-th feature, we define: \( f_r = [ f_{r1}, f_{r2}, \ldots, f_{rm} ]^T \), \( D = \text{diag}(S1) \), \( 1 = [1,1,\ldots,1]^T \), \( L = D - S \) where the matrix \( L \) is often called graph Laplacian. Let \( \bar{f}_r = f_r - \frac{g^T 1}{1^T 1} 1 \)
- Compute the Laplacian Score of the \( r \)-th feature as follows: 
\[
L_r = \frac{\bar{f}_r^T L \bar{f}_r}{\bar{f}_r^T D \bar{f}_r}.
\]

2.3. Fuzzy Kernel C-Means

2.3.1. Kernel function. We used a modified Fuzzy C-Means using Kernel function. Kernel method being used to map the non-linear input data space into a high dimensional feature space of the data. There are some Kernel functions that already known:

- RBF Kernel function : \( k_{ij} = k(x,y) = \exp\left(\frac{-\|x-y\|^2}{\sigma^2}\right) \)
- Polynomial Kernel function : \( k_{ij} = k(x,y) = (x^T y + 1)^d \)
- Linear Kernel function : \( k_{ij} = k(x,y) = x^T y \)

In this research, we used the RBF Kernel function to the classifier with \( \sigma = 0.05 \).

2.3.2. Fuzzy Kernel C-Means (FKCM). Basically, this modified classifier based on the standard Fuzzy C-Means (FCM). The difference between FKCM and FCM is on the distance function. The distance function on FCM Algorithm replaced by Kernel function (RBF function). So that, the objective function becomes:
\[
J = \min \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m d^2 (x_k, v_i) + \frac{\beta d^2 (s_k, v_i)}{a_{kj}} \tag{1}
\]

And for the membership function becomes:
\[ u_{ik} = \frac{1}{\sum_{l=1}^{c} \left( \frac{\beta_k d^2(v_i, v_l)}{a_{kl}} + d^2(v_i, v_l) \right)^{\frac{1}{m-1}}} \]  

(2)

where,

\[ v_i = \frac{\sum_{k=1}^{N} u_{ik}^m (x_k + \frac{\beta_k}{a_{kl}})}{\sum_{k=1}^{N} u_{ik}^m (1 + \frac{\beta_k}{a_{kl}})} \]  

(3)

Step 1 : Initialization
a. Matrix \( X \) with size \( n \times p \), where \( n \) is number of parameter;
b. Number of cluster \( c \);
c. Degree of fuzzy, \( m = 2 \);
d. Max iteration \( N \);
e. Small positive number \( \epsilon \) which is iteration stopping criteria;
f. Initial centroid by calculating the average value of each cluster

Step 2 : Update fuzzy membership degree of each data on each cluster using the equation (2)

Step 3 : Calculated centroid \( v_i \in V \) using equation (3). Where \( V = \{ v_1, v_2, ..., v_l \} \)

Step 4 : Determine iteration stopping criteria. If \( \Delta = \| v^t - v^{t-1} \| < \epsilon \), then iteration stops. Else then go to step 2.

2.4. The accuracy result
This formula is used to see the result of the accuracy of classification. The higher accuracy the more model classified the data correctly [6].

\[ \text{Accuracy} = \frac{\text{Number of True Prediction}}{\text{Number of test Data}} \times 100\% \]

3. Experiment results
For the result using the KDDCup 99 Dataset, there are 5 classes based on binary classification:

a. Normal
b. DoS
c. R2L
d. U2R
e. Probe

This dataset will be processed by LSFS to build a new data set/ an optimal dataset. After that, the optimal dataset will be processed by classifier FKCM to see the accuracy result of classification.

3.1. Feature selection result
On the table 2 shows the results of using feature selection for each experiment of classes. There are 5 features that selected by LSFS that shows for each experiment resulted different features and also different order of features. For example, in experiment Normal vs DoS resulted feature 6, feature 5, feature 1, feature 23, and feature 24 which based on table 1 are count feature, service feature, duration feature, logged_in feature, and dst_host_count feature, respectively.
Table 2. Laplacian Score Feature Selection (LSFS) results

| Experiment          | Selected Features |
|---------------------|-------------------|
| Normal vs DoS       | 6,5,1,23,24       |
| Normal vs Probe     | 6,5,2,33,32       |
| Normal vs U2R       | 5,6,1,33,32       |
| Normal vs R2L       | 6,5,1,33,23       |
| All: Normal, DoS,   | 6,5,1,23,24       |
| Probe, U2R, R2L     |                   |

Table 3. The result of IDS model using LSFS

| Experiment          | Number of Features | % Data Training | Accuracy (%) | Running Time |
|---------------------|--------------------|-----------------|--------------|--------------|
| Normal vs DoS       | 5                  | 90              | 97.50%       | 0.06         |
| Normal vs Probe     | 5                  | 90              | 72.00%       | 0.03         |
| Normal vs U2R       | 5                  | 50              | 75.00%       | 0.05         |
| Normal vs R2L       | 5                  | 30              | 91.43%       | 0.05         |
| All: Normal, DoS,   | 5                  | 90              | 60.00%       | 0.13         |
| Probe, U2R, R2L     |                    |                 |              |              |

We see on Normal vs DoS Experiment shows that the best accuracy resulting 97.50% with 90% data training, on Normal vs Probe Experiment shows that the best accuracy resulting 72% with 90% data training, on Normal vs U2R Experiment shows that the best accuracy resulting 75% with 50% data training, on Normal vs R2L Experiment shows that the best accuracy resulting 91.43% with 30% data training, and on All Classes Experiment shows that the best accuracy resulting 60% with 90% data training.

Table 4. The result of IDS model without feature selection

| Data                | Number of Features | % Data Training | Accuracy (%) | Running Time |
|---------------------|--------------------|-----------------|--------------|--------------|
| Normal vs DoS       | 42                 | 80              | 96.25%       | 0.03         |
| Normal vs Probe     | 42                 | 90              | 80.00%       | 0.05         |
| Normal vs U2R       | 42                 | 90              | 95.00%       | 0.03         |
| Normal vs R2L       | 42                 | 10              | 76.11%       | 0.09         |
| All: Normal, DoS,   | 42                 | 90              | 67.06%       | 0.09         |
| Probe, U2R, R2L     |                    |                 |              |              |

On this IDS model without using LSFS on it, we see on Normal vs DoS Experiment shows that the best accuracy resulting 96.25% with 80% data training, on Normal vs Probe Experiment shows that the best accuracy resulting 80% with 90% data training, on Normal vs U2R Experiment shows that the best accuracy resulting 95% with 90% data training, on Normal vs R2L Experiment shows that the best accuracy resulting 76.11% with 10% data training, and on All Classes Experiment shows that the best accuracy resulting 67.06% with 90% data training.

3.2. Accuracy result of model using feature selection

After we have those selected features on each experiment which means we have new data set that have small number of features for each experiment, we tried to classify those of new data set for each experiment using FKCM method with kernel RBF and parameter 0.05 and see the best accuracy resulted for each experiment showed by table 3.

3.3. Accuracy result of model without feature selection

On the other hand, the IDS model using FKCM without feature selection showed on table 4 displays the accuracy resulted for each experiment.
4. Conclusions
As we can see the results from table 3 and table 4, if we compare the accuracy results between those tables, the overall result shows that the Intrusion Detection System Model without LSFS has a better accuracy result rather than with LSFS on model. It showed on some experiments (Normal vs Probe, Normal vs U2R, and All Classes) from table 4 displays better accuracy compare to experiments accuracy result from table 3 that related on each and the comparison accuracy results between table 4 and table 3 are 80 % vs 72 %, 95 % vs 75 %, and 67.06 % vs 60 %, respectively. So, in this write we can say that IDS Model using FKCM as a classifier has better accuracy compare to IDS Model using the combination FKCM and LSFS as a classifier and feature selection, respectively.

References
[1] Hettich S and Bay S D 1999 The UCI KDD Archive (Irvine: University of California) KDD Cup 1999 Data
[2] Zhang D, Chen S and Zhou Z H 2008 Pattern Recogn. 41 1440–51
[3] Zhao Z and Liu H 2007 Proc. 24th International Conference on Machine Learning (ICML07), ed Z Ghahramani June 2007 Corvalis (New York: Association of Computing Machinery) pp 1151–7
[4] He X, Cai D and Niyogi P 2005 Proc. 18th International Conference on Neural Information Processing Systems (Advances in Neural Information Processing Systems vol 18) ed Y Weiss et al. (Cambridge: MIT Press) pp 507–14
[5] Rustam Z and Talita A S 2015 J. Theor. Appl. Inf. Technol. 81 161–5
[6] Rachman A A and Rustam Z 2016 Proc. 12th International Conference on Mathematics, Statistics, and Their Applications (ICMSA) 4-6 October 2016 Banda Aceh, ed S Arif and R Otavia (Banda Aceh: Universitas Syiah Kuala) MA-07 pp 31–4