An Analysis of Multiclass Imbalanced Data Problem in Machine Learning for Network Attack Detections

Hui Fern Soon¹², Amiza Amir¹² and Saidatul Norlyana Azemi¹²

¹ School of Computer & Communication Engineering, Universiti Malaysia Perlis (UniMAP), Malaysia
² Advanced Communication Engineering (ACE) Centre of Excellence, UniMAP, Malaysia

amizaamir@unimap.edu.my

Abstract. In the current trend, machine learning has been used widely for network attack detection. The performance of a machine learning model depends on its training dataset and the dataset distribution. Network attack detection is one of the problems that usually suffer from the imbalanced data distribution. However, the effect of this imbalanced data is generally neglected by researchers. Therefore, in this research, we studied the impact of an imbalanced multiclass dataset to the machine learning performance. Five state-of-the-art machine learning algorithms were used in this study, and the classifiers that can classify the minority class or the majority class instances accurately were also identified. In this research, the performances of these classifiers were evaluated by using the performance metrics: true positive rate, false positive rate, precision, F-measure, ROC area, and classification accuracy. The results show that the J48 classifier outperforms the other four classifiers in every aspect. Besides, Naïve Bayes, J48, has also worked as the best classifier that able to classifies the instances of the minority class accurately.

1. Introduction

Network Attack Detection is a security method used to track the suspicious and unusual activities of a network such as backdoor, viruses, worms, etc. Machine learning can be used to differentiate whether the network data is normal or anomalous data.

There are many different types of network attacks, and the occurrence of these unexpected and malicious activities might be very different. If the types of these malicious activities are managed to detect correctly, the administrator can take immediate action to solve the problem accordingly. However, from [1]–[8] we notice that many researchers only experiment on evaluating the accuracy of classification of the machine learning algorithms on the networks attacks data without taking the imbalanced occurrence of these network attacks into account. With imbalanced data occur, the classification might lead to the wrong prediction due to overfitting as the minority class used to be ignored in the imbalanced datasets. Despite the high accuracy of the summarized results, none of these works look at the detailed level of how the algorithms deal with minority classes. The problem of the multiclass imbalanced problem is not a new issue in machine learning. However, studies that use machine learning for network attack detection often ignore this issue. Hence, the analysis to examine
this problem in the network attack detection domain is important to improve the detection of minority classes.

Therefore, the purpose of this research is to analyze and evaluate the performance of state-of-the-art (SOTA) machine learning algorithms on the multiclass imbalanced network attack detection dataset (UNSW-NB15). The contributions of this research are:

- We analyze which machine learning algorithms are more suitable to be used on the multiclass imbalanced network attack detection dataset (UNSW-NB15).
- We identify which machine learning algorithms perform better in classifying the instances of majority class or minority class and achieved high accuracy.

Generally, KDD CUP 1999 dataset is used for network anomaly detection by the researchers. Yet, [4], [6], [8] have stated that this dataset contains a lot of drawbacks. Therefore, in this research, the UNSW-NB15 dataset [9], [10], which has overcome the disadvantage of the KDD CUP 1999 dataset is being used.

2. Related Works

Without considering the imbalanced problem in the network anomaly detection dataset, many researchers have experimented on searching and investigating which machine learning (ML) is able to detect the network anomaly data perfectly. Random Forest (RF) has found to be performed well in detecting network attack data than Support Vector Machine (SVM), Logistic regression (LR), stochastic Gradient Descent, and Sequential Model classifier in [3]. Based on [2], the researchers have found out that REP-Tree and J48 classifier have achieved excellent results in every aspect. However, in [5] decision tree classifier is said to overfit, and it has been worked well in training data but poor in test data in this research. It also found that the Linear algorithm has provided a poor accuracy in detecting the network attack data. [7] have also proposed an algorithm named as Online Locally Weighted Projection Regression (OLWPR) for network attack detection. This proposed algorithm has performed magnificently in accuracy rate, average percentage error, average RMSE, and F1 score than Artificial Neural Network (ANN), SVM, Decision Tree, LR, and RF.

To improve the accuracy of detecting the network attack data, ensemble machine learning algorithms have also been used by some researchers. [4] have created an ensemble machine learning based on Fuzzy C-Mean (FCM), Naïve Bayes (NB), KMeans, Radial Basis Function (RBF), K-Nearest Neighbours (KNN) and SVM. This ensemble ML has provided a good result in accuracy, recall, and precision, yet its performance still does not surpass RBF. Not every ensemble ML algorithms are able to provide excellent results in detecting anomaly data. As shown in [2], the results show that boosting and bagging ensemble classifiers do not have the ability to improve classification accuracy.

Other than ML algorithms, a deep learning approach can also be used to detect network attack data. In [1] and [8], the deep learning approach has shown very good accuracy in detecting the network anomaly data. The deep learning approach in [8] which combines the supervised and unsupervised ML methods, has higher accuracy than SVM, decision tree, Random Forest, KNN, NB, and multi-layer perceptron with feature extraction.

[6] did take the imbalance classification of network attack data into consideration. This research has worked on improving the detection of the overall accuracy and the low occurrence network attack (minority class) which is the U2R attack in the NSL-KDD Cup99 dataset using four feature subset selection techniques. It also applied three different approaches: classification, data clustering, and anomaly-based approach. However, it only employs two rule-based classifiers (Decision Table and PART rule-based classifier) to evaluate the performance on the reduced feature set.
3. Multiclass Imbalanced Dataset

This UNSW-NB15 dataset contains a total of 257673 data instances with 44 attributes (Table 1). These data instances consist of 10 different classes, where one out of these 10 classes are for the normal network data and the remaining 9 classes are the anomalous network data.

**Table 1. Attributes of UNSW-NB15 Dataset.**

| No. | Attributes | No. | Attributes |
|-----|------------|-----|------------|
| 1   | id         | 23  | dtrcpb     |
| 2   | dur        | 24  | dwin       |
| 3   | proto      | 25  | tcprrtt    |
| 4   | service    | 26  | synack     |
| 5   | state      | 27  | ackdat     |
| 6   | spkts      | 28  | smean      |
| 7   | dpkts      | 29  | dmean      |
| 8   | sbytes     | 30  | trans_depth|
| 9   | dbytes     | 31  | response_body_len |
| 10  | rate       | 32  | ct_srv_src |
| 11  | sttl       | 33  | ct_state_ttl|
| 12  | dttl       | 34  | ct_dst_ltm |
| 13  | sload      | 35  | ct_src_dport_ltm |
| 14  | dload      | 36  | ct_dst_sport_ltm|
| 15  | sloss      | 37  | ct_dst_srch_ltm|
| 16  | dloss      | 38  | is_ftp_login|
| 17  | sinpkt     | 39  | ct_ftp_cmd |
| 18  | dinkpt     | 40  | ct_flw_http_mthd|
| 19  | sjit       | 41  | ct_src_ltm |
| 20  | djit       | 42  | ct_srv_dst |
| 21  | swin       | 43  | is_sm_ips_ports|
| 22  | stepb      | 44  | attack_cat |

Table 2 below shows the number of instances contains in each class. The dataset is considered as imbalanced since the number of instances for each class is uninform. From 257673 instances, 36% of them are normal class and 22% are generic class. Other attacks such as backdoor, analysis, shellcode, works, and reconnaissance have a very low number of samples (1% or less). These attacks are usually will be very difficult to be classified due to insufficient training samples.

**Table 2. Numbers of instances in each class in UNSW-NB15 Dataset.**

| Classes            | Number of instances | % of instances |
|--------------------|---------------------|----------------|
| Normal             | 93000               | 36.09          |
| Backdoor           | 2329                | 0.90           |
| Analysis           | 2677                | 1.04           |
| Fuzzers            | 24246               | 9.41           |
| Shellecode         | 1511                | 0.59           |
| Reconnaissance     | 13987               | 0.05           |
| Exploits           | 44525               | 17.28          |
| Dos                | 16353               | 6.35           |
| Worms              | 174                 | 0.07           |
| Generic            | 58871               | 22.85          |
4. Experiment

4.1. Machine Learning algorithms

Based on [11], in machine learning, classification is a prognostic method where the class of the instances is decided. We aim to analyze the behavior of different types of machine learning when dealing with a highly imbalanced dataset. Hence, we select five different machine learning algorithms from four different families: Rule-based, Bayesian, Instance-based, and Decision Tree.

This research undergoes three phases. First, we load the whole dataset used in this research into WEKA for classification purposes. Then the machine learning algorithms are used to train and learn the pattern of the data instances. This step is crucial for the purpose to evaluate the machine learning classifiers' performances. Next, 10-fold cross-validation is then applied, and lastly, the testing stage is going through by the built classifier. After completing these phases, the classification performance of the machine learning algorithms over the UNSW-NB15 dataset can now be measured.

4.1.1. Rule-based Classifier

For classification purposes, Rule-based classifiers follow a set of IF-THEN rules [6], [12]. The structure of the IF-THEN rule is: IF Condition (conjunction of attributes) – THEN conclusion (class prediction). Based on [12], rule-based classifiers have the ability to classified the new instances rapidly. There are many different types of rule-based classifiers, however, in this research, we will focus on ZeroR and OneR classification methods.

- ZeroR: ZeroR is an uncomplicated rule-based algorithm that focuses on the target variable and refrains all predictors [11], [13], [14]. It is a baseline classifier that always classified the data instances to the majority class [14].
- OneR: OneR classifier is a straightforward algorithm that measures the features based on the error rate. It generates a set of rules and selects the rule with the lowest error [15].

4.1.2. Bayesian: Naïve Bayes (NB)

Naïve Bayes is a classifier based on Bayes Theorem [11], [16], [17]. It is one of the easy and widely used classifiers for classification purposes. It works regarding conditional probability. It assigned a probability to each class variable and the class variables with the highest score will be chosen [11]. Besides, NB can be applied to any size of datasets and provide a good performance [11], [18]. It also manages to handle binary and multiclass classification problems too [18].

4.1.3. Instance-Based: k-NN

k-NN (k-Nearest Neighbor) classifier is an instance-based and lazy based classifier, [11], [19]–[21]. The task in searching the nearest neighbors manage to speed up by applying this algorithm. When there are new data instances, these instances will be categories under the class with topmost k nearest neighbor, where k will never be a non-positive integer [11], [21]. In this research, the k value was set to 10 as it obtains the highest classification accuracy.

4.1.4. Decision Tree: J48

J48, a decision tree classifier that generates a binary tree is the C4.5 algorithm’s open-source Java implementation in WEKA software [21]–[23]. For the purpose to illustrate the classification process, the decision tree is created by this classifier by implementing a greedy search and top-down through each probable branch [22]. Based on [21], this classifier is applied in order to avoid overfitting and achieve supreme accuracy on training data. In building a decision tree, J48 acknowledges both categorical and continuous attributes. Moreover, tree pruning can also be done in order to trim the size of the decision tree and to minimize the error of misclassification [21].
4.2. Weka
A data mining tool named as WEKA (Waikato Environment for Knowledge Analysis) which developed by the University of Waikato in New Zealand in 1993 [6], [19] was used to carry out this research. Weka is a free software and it can be run under various types of platforms such as Macintosh operating system, Windows, and Linux. It is user friendly as one can easily apply the state-of-the-art machine learning algorithms on a dataset and examine its performance without the need of any program code. Besides, WEKA also contains the methods for regression, clustering, attribute selection, classification, and association rule mining. Other than analysis data, it can also be applied for forecasting and prediction purposes [19].

4.3. Performance Metrics
There are a lot of parameter metrics out there that can be used to measure evaluate the performance of a classifier. In this research, we use the parameter metrics: TPR (true positive rate), FPR (false positive rate), accuracy, precision, F-measure, and ROC Area to evaluate the performance of the state-of-the-art machine learning algorithms on UNSW-NB15 dataset. In order to measure these parameters, an unambiguous perspective of the classifier’s performance is required and hence confusion matrix is needed to be tabulated. Usually, a confusion matrix as shown in Table 3 are tabulated; however, this confusion matrix is used for binary class classification. As the UNSW-NB15 dataset used in this research is a multiclass imbalanced dataset, therefore a confusion matrix as shown in Table 4 will be tabulated.

**Table 3.** Confusion Matrix for binary classification problem.

| Class        | Predicted |          |
|--------------|-----------|----------|
|              | Positive  | Negative |
| Actual       | Positive  | TP (True Positive) | FN (False Negative) |
|              | Negative  | FP (False Positive) | TN (True Negative) |

**Table 4.** Confusion Matrix for multiclass classification.

| Class 1 | Class 2 | Class 3 | … | Class n-1 | Class n | Predicted/Actual |
|---------|---------|---------|---|-----------|---------|------------------|
| Accurate| Accurate| Accurate|   | Accurate  | Accurate | Class 1          |
|         |         |         |   |           |         | Class 2          |
|         |         |         |   |           |         | Class 3          |
|         |         |         |   |           |         | …                |
|         |         |         |   |           |         | Class n-1        |
|         |         |         |   |           |         | Class n          |

TP: positive class instances that assign accurately as positive
TN: negative class instances that assign accurately as negative
FP: negative class instances that mistakenly assign as positive
FN: positive class instances that mistakenly assign as negative

Based on the correctly classified instances shown in Table 4, the percentage of accuracy of the SOTA machine learning algorithms used on the UNSW-NB15 dataset in this research can be calculated by using (1).

\[ \text{Accuracy} = \frac{\text{Number of instances that classified correctly}}{\text{Total numbers of instances}} \times 100 \] (1)

True positive rate (TPR) which is also known as sensitivity, is equivalent to another performance metric, Recall [24]. It is the ratio of correctly classified positive class instances over the total number
of positive instances. The classifier is said to be performed excellently when the TPR value of the classifier is close to 1 [25]. The formula of TPR is shown in (2) below.

Other than calculating the TPR value of a classifier, the performance of a machine learning algorithm can also be measure by using False Positive Rate (FPR). It is defined as the fraction of FP instances over all negative instances. (3) show the formula for FPR. A classifier is said to be performed magnificently when the FPR value of the classifier is very low [25].

\[
TPR = \frac{TP}{TP+FN} \quad (2)
\]
\[
FPR = \frac{FP}{FP+TN} \quad (3)
\]

Precision, F-measure, and ROC Area are also being used in this research to evaluate the performance of the classifiers used in this research on the UNSW-NB 15 dataset. Precision is the ratio of the TP instances over all instances, which have been predicted as positive. Its formula has shown in (4). Based on [26], F-measure combines both Recall and Precision in order to express their tradeoff. Parameter \( \beta \) which generally set as 1 is used to modify the relative importance of precision and recall.

\[
\text{Precision} = \frac{TP}{TP+FP} \quad (4)
\]
\[
F\text{-measure} = \frac{(1+\beta^2)(\text{Recall})(\text{Precision})}{\beta^2(\text{Recall})+(\text{Precision})} \quad (5)
\]

Apart from the parameter metrics discussed above, the ROC area of being used in this research. It refers to the area under a curve created by plotting the graph of TPR against FPR at different threshold [27].

5. Results and Discussion

Table 5 shows the performances of the TPR, FPR, Precision, F-measure, and ROC area for the SOTA classifiers used in this research. Based on this table, it shows that all classifiers have given an excellent TPR value (which is above 0.9) for Generic class except for ZeroR classifier. Although there is a total of 10 classes in the UNSW-NB15 dataset, ZeroR only gives a TPR and FPR value to the Normal class, which is very different from another rule-based classifier OneR. This is because ZeroR yields the highest majority class. It can be visualized clearly in Table 6 where all instances have been classified to the Normal class (the highest majority class) and this has caused the overall classification accuracy of ZeroR to be less than 50% (Figure 1).

Among all the 10 classes in the UNSW-NB15 dataset, the Analysis attack contains the lowest average TPR value which is 0.0896 follows by Dos attack class (0.1182). This means that these classes are not really recognized by all the SOTA classifiers in the UNSW-NB15 dataset. Besides, this table also shows that ZeroR classifier gives the highest average FPR value among all the classifiers, and this has cause Normal class to be the worst case for classifying on average. By ignoring the FPR of ZeroR for Normal class, the worst-case to classify on average goes to Exploits attack with an average FPR value of 0.0776. As shown in Table 2, there are a total of 44524 instances for Exploits attack out of 257673 instances, which is the second-highest network attack class in the UNSW-NB15 dataset. If this attack is failed to detect or classifies well, administrators might face the problem in solving it immediately and bad things might happen.

As stated in Section 4.3, a classifier with the TPR value close to 1 or the FPR value close to 0 can be known as a classifier that well performed. Based on Table 5, we can see that the J48 classifier not only achieve the lowest FPR value, it also achieved the highest value in TPR, Precision, F-measure and ROC area among all the SOTA classifiers used in this research. Therefore, it can be concluded that J48 has acted as the best classifier that able to classifies excellently among all the classifiers used
in this research. On the other hand, ZeroR which contains the highest FPR value and lowest value in TPR Precision, F-measure, and ROC area can be said as the worst classifier in this research.

Figure 1. Percentage of Classification Accuracy of SOTA Classifiers for Imbalanced Multiclass UNSW-NB15 dataset.

In an imbalanced multiclass dataset, the minority class instances normally hold more important information than the majority class instances. Yet, researchers normally only focus on improving the overall classification accuracy. A classifier that able to obtain a high overall classification accuracy does not mean that it is also able to classify the instances in the minority class well. Table 6 to Table 10 shows the confusion metrics of ZeroR, OneR, NB, k-NN, and J48 on the UNSW-NB15 dataset. Based on these tables, NB and J48 have shown to be the classifiers that able to classifies the minority class instances well. However, from these tables we also able to notice that although NB manages to classify most of the minority class accurately, it is poor in classifies the majority class instances, and this has caused NB to contain a low percentage of overall classification accuracy as shown in figure 1.

On the other hand, although OneR and k-NN algorithms have shown a high classification accuracy in figure 1, Table 7 and Table 9 have shown that these classifiers do not perform well in classifying the instances for minority class as compared to the NB classifier. However, they performed quite well in classifying the majority class instances.

Among all the classifiers used in this research towards the imbalanced multiclass UNSW-NB15 dataset, J48 is the best classifier that performed well in classifying the instances for minority class and majority class. It has also achieved the highest percentage of classification accuracy (87.56%) among the ZeroR, OneR, NB, k-NN, and J48 machine learning algorithms for the UNSW-NB15 dataset in this research.

Table 5. Comparison of SOTA classifiers performance on TPR, FPR, Precision, Recall and ROC area for Imbalanced Multiclass UNSW-NB15 dataset.
|                | Exploits | 0.500 | 0.585 | 0.514 | 0.761 | 0.519 | 0.828 | 0.813 |
|----------------|----------|-------|-------|-------|-------|-------|-------|-------|
|                | Dos      | 0.500 | 0.024 | 0.009 | 0.200 | 0.358 |
| Average        |          | 0.1   | 0.4268| 0.4172| 0.4396| 0.6287|
|                | FPR      | Normal| 1     | 0.093 | 0.009 | 0.035 | 0.005 |
|                |          | 0.581 | 0.0313| 0.0505| 0.0225| 0.0145|
|                | Precision| Normal| 0.361 | 0.842 | 0.972 | 0.959 | 0.991 |
|                |          | 0.530 | 0.859 | 0.736 | 0.964 | 0.992 |
|                |          | 0.500 | 0.891 | 0.906 | 0.973 | 0.997 |
|                |          | 0.500 | 0.519 | 0.861 | 0.496 | 0.824 |
| Average        |          | 0.053 | 0.4432| 0.2683| 0.4737| 0.6591|
|                | ROC Area | Normal| 0.500 | 0.519 | 0.861 | 0.496 | 0.824 |
|                |          | 0.500 | 0.500 | 0.879 | 0.568 | 0.844 |
|                |          | 0.500 | 0.500 | 0.879 | 0.568 | 0.844 |
|                |          | 0.500 | 0.842 | 0.791 | 0.775 | 0.956 |
| Average        |          | 0.493 | 0.615 | 0.874 | 0.569 | 0.889 |
|                |          | 0.500 | 0.896 | 0.984 | 0.987 | 0.996 |
### Table 6. Confusion Matrix of ZeroR for Imbalanced Multiclass UNSW-NB15 dataset.

|   | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   | Class   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|
| 93000 | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | a= Normal |
| 2329   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | b= Backdoor |
| 2677   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | c= Analysis |
| 24246  | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | d= Fuzzers |
| 1511   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | e= Shellcode |
| 13987  | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | f= Reconnaissance |
| 44525  | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | g= Exploits |
| 16353  | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | h= Dos |
| 174    | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | i= Worms |
| 58871  | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | j= Generic |

### Table 7. Confusion Matrix of OneR for Imbalanced Multiclass UNSW-NB15 dataset.

|   | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   | Class   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|
| 81424 | 15  | 1   | 4561| 138 | 73  | 6705| 73  | 9   | 1   | 1   | a= Normal |
| 124   | 88  | 0   | 41  | 1   | 1   | 2072| 1   | 0   | 1   |    | b= Backdoor |
| 140   | 0   | 9   | 20  | 0   | 0   | 2515| 2   | 0   | 0   |    | c= Analysis |
| 8444  | 22  | 0   | 9048| 136 | 23  | 6503| 56  | 8   | 6   |    | d= Fuzzers |
| 578   | 5   | 0   | 242 | 329 | 36  | 265 | 43  | 0   | 13  |    | e= Shellcode |
| 407   | 3   | 0   | 131 | 7   | 9578| 3834| 25  | 1   | 1   |    | f= Reconnaissance |
| 4354  | 2   | 3   | 1884| 76  | 83  | 37882| 189 | 28  | 24  |    | g= Exploits |
### Table 8. Confusion Matrix of NB for Imbalanced Multiclass UNSW-NB15 dataset.

|   | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   | Class   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|
| a | 55087 | 4086 | 5703 | 6570 | 15048 | 941 | 4475 | 328 | 222 | 540 | Normal  |
| b | 0 | 1599 | 67 | 13 | 548 | 8 | 19 | 3 | 26 | 46 | Backdoor |
| c | 14 | 1646 | 549 | 17 | 346 | 4 | 19 | 2 | 34 | 46 | Analysis |
| d | 128 | 3615 | 77 | 5342 | 12889 | 1456 | 180 | 16 | 67 | 476 | Fuzzers  |
| e | 0 | 4 | 0 | 25 | 1474 | 0 | 0 | 0 | 0 | 8 | Shellcode |
| f | 5 | 1676 | 62 | 195 | 11843 | 57 | 29 | 7 | 17 | 96 | Reconnaissance |
| g | 1264 | 12264 | 2021 | 1485 | 9852 | 763 | 13982 | 217 | 2438 | 239 | Exploits  |
| h | 86 | 9615 | 636 | 313 | 3557 | 261 | 1045 | 142 | 525 | 173 | Dos      |
| i | 1 | 1 | 2 | 9 | 122 | 3 | 2 | 0 | 34 | 0 | Worms   |
| j | 63 | 330 | 46 | 126 | 559 | 94 | 279 | 23 | 275 | 57076 | Generic |

### Table 9. Confusion Matrix of k-NN for Imbalanced Multiclass UNSW-NB15 dataset.

|   | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   | Class   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|
| a | 90033 | 0 | 14 | 2335 | 12 | 257 | 327 | 10 | 0 | 12 | Normal  |
| b | 43 | 65 | 46 | 286 | 0 | 94 | 1406 | 387 | 0 | 2 | Backdoor |
| c | 160 | 83 | 190 | 183 | 0 | 2 | 1634 | 424 | 0 | 1 | Analysis |
| d | 3535 | 29 | 13 | 16205 | 61 | 1281 | 2521 | 580 | 0 | 21 | Fuzzers  |
Table 10. Confusion Matrix of J48 for Imbalanced Multiclass UNSW-NB15 dataset.

| Class | a | b | c | d | e | f | g | h | i | j |
|-------|---|---|---|---|---|---|---|---|---|---|
| Normal | 92279 | 2 | 13 | 496 | 26 | 21 | 130 | 27 | 0 | 6 |
| Backdoor | 457 | 30 | 188 | 15 | 29 | 1194 | 410 | 0 | 3 | 0 |
| Analysis | 134 | 461 | 213 | 0 | 3 | 1330 | 510 | 0 | 3 | 0 |
| Fuzzers | 20563 | 127 | 44 | 2074 | 727 | 4 | 32 | 0 | 0 | 0 |
| Shellcode | 1048 | 20 | 176 | 54 | 2 | 14 | 0 | 0 | 0 | 0 |
| Reconnaissance | 10718 | 2325 | 792 | 1 | 4 | 0 | 0 | 0 | 0 | 0 |
| Exploits | 36186 | 5331 | 44 | 233 | 0 | 0 | 0 | 0 | 0 | 0 |
| Dos | 5783 | 4 | 81 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Worms | 93 | 2 | 0 | 0 | 7 | 2 | 1 | 63 | 6 | 0 |
| Generic | 58026 | 2 | 0 | 0 | 7 | 2 | 1 | 63 | 6 | 0 |

6. Conclusion
In conclusion, the J48 classifier has outperformed the other four classifiers that have been used in this research in every aspect (TPR, FPR, Precision, F-Measure, ROC Area, and Classification Accuracy) for multiclass imbalanced UNSW-NB15 datasets. In order, to classify the instances for the minority class on network anomaly datasets, it is preferable to use the NB and J48 classifier as these two classifiers contain the ability to classify the minority class instances accurately. However, to classify
the instances of majority class, the J48 classifier followed by k-NN and OneR classifier can be used. Other than that, ZeroR classifier is not suitable to be used to classify the network anomaly data, as it does not contain any predictive strength. This is because ZeroR classifier used to yields all the instances of a dataset to the highest majority class instances.

The output from this research is useful for our future work to devise a new strategy to deal with the multiclass imbalanced problem. Different classifiers act differently when dealing with minority or majority class.

Acknowledgments
Fundamental Research Grant Scheme (FRGS900300655 1/2018) has support this work. We are grateful to our colleagues who have provided expertise that has greatly assisted the research, although they may or may not agree with all of the interpretations contained in this paper. We are also thankful to Advanced Communication Engineering Centre (ACE) and School of Computer and Communication Engineering, Universiti Malaysia Perlis, which provided laboratory facilities during this work.

7. References
[1] R. K. Malaiya, D. Kwon, S. C. Suh, H. Kim, I. Kim, and J. Kim, “An Empirical Evaluation of Deep Learning for Network Anomaly Detection,” IEEE Access, vol. 7, pp. 140806–140817, 2019, doi: 10.1109/ACCESS.2019.2943249.
[2] M. Belouch and S. El Hadaj, “Comparison of ensemble learning methods applied to network intrusion detection,” ACM Int. Conf. Proceeding Ser., pp. 7–10, 2017, doi: 10.1145/3018896.3065830.
[3] F. Yihunie, E. Abdelfattah, and A. Regmi, “Applying Machine Learning to Anomaly-Based Intrusion Detection Systems,” 2019 IEEE Long Isl. Syst. Appl. Technol. Conf. LISAT 2019, 2019, doi: 10.1109/LISAT.2019.8817340.
[4] M. Zaman and C. H. Lung, “Evaluation of machine learning techniques for network intrusion detection,” IEEE/IFIP Netw. Oper. Manag. Symp. Cogn. Manag. a Cyber World, NOMS 2018, pp. 1–5, 2018, doi: 10.1109/NOMS.2018.8406212.
[5] A. N. Sokolov, I. A. Pyatnitsky, and S. K. Alabugin, “Research of Classical Machine Learning Methods and Deep Learning Models Effectiveness in Detecting Anomalies of Industrial Control System,” Proc. - 2018 Glob. Smart Ind. Conf. GloSIC 2018, pp. 1–6, 2018, doi: 10.1109/GloSIC.2018.8570073.
[6] S. Bahl and S. K. Sharma, “Improving classification accuracy of intrusion detection system using feature subset selection,” Int. Conf. Adv. Comput. Commun. Technol. ACCT, vol. 2015-April, pp. 431–436, 2015, doi: 10.1109/ACCT.2015.137.
[7] I. G. A. Poornima and B. Paramasivan, “Anomaly detection in wireless sensor network using machine learning algorithm,” Comput. Commun., vol. 151, no. December 2019, pp. 331–337, 2020, doi: 10.1016/j.comcom.2020.01.005.
[8] H. N. Viet, L. L. T. Trang, Q. Nguyen Van, and S. Nathan, “Using deep learning model for network scanning detection,” ACM Int. Conf. Proceeding Ser., pp. 117–121, 2018, doi: 10.1145/3233347.3233379.
[9] N. Moustafa and J. Slay, “UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” 2015 Mil. Commun. Inf. Syst. Conf. MilCIS 2015 - Proc., pp. 1–6, 2015, doi: 10.1109/MilCIS.2015.7348942.
[10] N. Moustafa and J. Slay, “The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set,” Inf. Secur. J., vol. 25, no. 1–3, pp. 18–31, 2016, doi: 10.1080/19393555.2015.1125974.
[11] C. Sugetha, L. Karunya, E. Prabhavathi, and P. K. Sujatha, “Performance Evaluation of Classifiers for Analysis of Road Accidents,” 2017 9th Int. Conf. Adv. Comput. ICoAC 2017, pp. 365–368, 2017, doi: 10.1109/ICOAC.2017.8441188.
[12] S. C. Ganiachchi, “Creation of Semantic Location Profiles using Bayes, Rule-Based, Trees and Meta Classification Approaches,” 2017 13th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Discov., pp. 1604–1609, 2017.
[13] A. U. B, P. Nanda, X. He, and K. R. Choo, of Facebook Messenger Mobile Data using WEKA, vol. 1. Springer International Publishing.
[14] H. B. Mirza, “Classifier Tools: A Comparative Study,” 2018 Second Int. Conf. Intell. Comput. Control Syst., no. Iccs, pp. 1543–1547, 2018.
[15] K. Mishra and R. Rani, “Churn prediction in telecommunication using machine learning,” 2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput. ICECDS 2017, no. 2012, pp. 2252–2257, 2018, doi: 10.1109/ICECDS.2017.839853.
[16] V. Mathew, A. M. Chacko, and A. Udhayakumar, “Prediction of suitable human resource for replacement in skilled job positions using Supervised Machine Learning,” Proc. 2018 8th Int. Symp. Embed. Comput. Syst. Des. ISED 2018, pp. 37–41, 2018, doi: 10.1109/ISED.2018.8704120.
[17] R. Saravanan and P. Sujatha, “A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification,” Proc. 2nd Int. Conf. Intell. Comput. Control Syst. ICICCS 2018, no. Iccs, pp. 945–949, 2019, doi: 10.1109/ICCONS.2018.8663155.
[18] S. Ray, “A Quick Review of Machine Learning Algorithms,” Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Prespectives Prospect. Com. 2019, pp. 35–39, 2019, doi: 10.1109/COMITCon.2019.8862451.
[19] E. Frank, M. A. Hall, and I. H. Witten, Data Mining: Practical Machine Learning Tools and Techniques, Fourth Ed. 2016.
[20] S. Dzeroski, Data Mining. 2008.
[21] S. Rahman, “Carbon Emission Measurement In Improved Cook Stove Using Data Mining,” 2017 Int. Conf. Electr. Comput. Commun. Eng., pp. 83–86, 2017, doi: 10.1109/ECACE.2017.7912884.
[22] S. Asha Kiranmai and A. Jaya Laxmi, “Data mining for classification of power quality problems using WEKA and the effect of attributes on classification accuracy,” Prot. Control Mod. Power Syst., vol. 3, no. 1, 2018, doi: 10.1186/s41601-018-0103-3.
[23] R. A. Omar and A. Tjahyanto, “Evaluation of the performance of a machine learning algorithms in Swahili-English emails filtering system relative to Gmail classifier,” 2018 Int. Conf. Inf. Commun. Technol. ICOIACT 2018, vol. 2018-Janua, pp. 266–269, 2018, doi: 10.1109/ICOIACT.2018.8350713.
[24] H. Kaur, H. S. Pannu, and A. K. Malhi, “A systematic review on imbalanced data challenges in machine learning: Applications and solutions,” ACM Comput. Surv., vol. 52, no. 4, 2019, doi: 10.1145/3343440.
[25] M. Nawir, A. Amir, N. Yaakob, and O. N. G. B. I. Lynn, “MULTI-CLASSIFICATION OF UNSW-NB15 DATASET FOR,” vol. 96, no. 15, pp. 5094–5104, 2018.
[26] T. Zhu, Y. Lin, and Y. Liu, “Synthetic minority oversampling technique for multiclass imbalance problems,” Pattern Recognit., vol. 72, pp. 327–340, 2017, doi: 10.1016/j.patcog.2017.07.024.
[27] S. S. Patil and S. P. Sonavane, “Enriched over-sampling techniques for improving classification of imbalanced big data,” Proc. - 3rd IEEE Int. Conf. Big Data Comput. Serv. Appl. BigDataService 2017, pp. 1–10, 2017, doi: 10.1109/BigDataService.2017.19.