Received: December 6, 2019


INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

# Analyzing the Performance of Intrusion Detection Model Using Weighted One-Against-One Support Vector Machine and Feature Selection for Imbalanced Classes

**Bambang Setiawan[1]*** **Supeno Djanali[1]** **Tohari Ahmad[1]**

*Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
* Corresponding author's Email: setiawan@is.its.ac.id

**Abstract:** Imbalanced class is one of the main problems for intrusion detection models that use machine learning methods. The classifiers generally are designed to minimize the global error rates, which have not considered the condition of imbalanced class. The amount of training data for each type of imbalanced attack can cause those intrusion detection models to have high accuracy but can also lead to difficulty in identifying minority class attacks. In this research, we propose an intrusion detection model using a combination of the feature selection method for imbalanced class and weighted support vector machine classifier. We apply a composite performance index in the features selection and the optimization of the weight of the minority class. The experimental result using the NSL-KDD dataset shows that this model produces overall accuracy, sensitivity, and specificity that reaches more than 99%, with false alarms below 0.5% and false-negative rates below 0.7%. The sensitivity of the U2R and R2L classes are 56% and 92%.

**Keywords:** Feature selection, Imbalanced class, Intrusion detection, Network security, Weighted support vector machine.

## 1. Introduction

Intrusion detection systems (IDS) dynamically monitor system activity in a certain environment and decide whether an activity will be considered an attack. Many IDS models have been developed using machine learning and data mining methods. Imbalanced class is one of the main problems for IDSs that use machine learning and data mining methods. The amount of training data for each type of imbalanced attack can cause those IDSs to have high accuracy but can also lead to the difficulty in identifying all types of attacks.

IDS dataset, such as KDD Cup 1999 and NSL-KDD, has a very imbalanced number of attack instances [1]. The research in [2–5] is the example of IDS that have high accuracy but have difficulty in detecting minority classes in NSL-KDD datasets. Their overall accuracy is above 95%, but the detection accuracy for minority classes (U2R) is below 25%.

Support vector machines (SVM), like most classifiers, implicitly assumes that all classes have the same occurrences. SVM is also designed to maximize the whole classification accuracy. These conditions cause SVM to favor more the majority classes, which results in low sensitivity towards the minority classes [6].

Resampling is the rebalancing strategy commonly used to imbalanced classification. These methods include under-sampling the majority classes and over-sampling the minority classes that attempt to rebalance class distribution at the data level. However, the under-sampling method carries the risk of losing information, and the over-sampling approach has the potential to cause over-fitting because random replication of the minority class samples usually creates particular rules [6].

Cost-sensitive learning is a rebalancing strategy that can overcome the problems faced by the resampling method. This approach tries to balance class distribution at the algorithm level so that it does not change the condition of the sample data. This method considers the cost of misclassification and minimizes the total misclassification of costs.

The weighted-SVM (WSVM) is implemented cost-sensitive learning in SVM. Yang et al. [7] present that a WSVM that can rectify the outlier sensitivity in standard SVM for two-class data classification. In multiclass classification, Aburomman and Reaz in [8] show that a weighted one-against-rest SVM (WOAR-SVM) outperforms the one-against-rest SVM (OAR-SVM) and OAO SVM. However, their experimental results using the NSL-KDD dataset show that the accuracy of the U2R and R2L classes is under 40%.

In our previous research [9], we conduct a rebalancing strategy using a feature selection method for an imbalanced dataset. We use a modified-rank information gain feature selection (m-RIGFS) method to attempt to rebalance class. This strategy is combined with the log normalization and one-against-one SVM (OAO-SVM) with kernel parameter optimization. The experimental result shows that this approach can improve the sensitivity of the minority classes.

Therefore, the aim of this study is to construct IDS using a combination of WSVM with feature selection for imbalanced classes. We hypothesize that this approach can improve the detection of minority classes and maintain overall accuracy that remains above 99%.

In this research, we propose an IDS model using cost-sensitive learning that combines with a feature selection method for the imbalanced dataset. For cost-sensitive learning, we implement weighted one-against-one SVM (WOAO-SVM). While for the feature selection method, we use m-RIGFS [9] and propose the rank weighted information-gain feature selection method (RWIGFS).

There are three uniqueness of the proposed model compared to the existing IDS approach. First, this model uses the feature selection method for imbalanced classes to get a feature subset that better supports the detection of a minority class. We use two feature selection methods: the m-RIGFS and the RWIGFS. Second, we apply the composite performance index (CPI) in the selection of the n-best feature subset and the optimization of the weight of minority class in WSVM. The CPI is generated from accuracy in the overall class and sensitivity in the minority classes using multiple attribute decision making (MADM) method. Third, this model integrates log normalization, feature selection method for imbalanced classes, and the WSVM classifier.

The rest of this paper is organized as follows. Section 2 describes the related work. In Section 3, we present the proposed work. In the next section, we explain and discuss the implementation results. Finally, we give a conclusion in Section 5.

## 2. Related work

Many models of IDS have been evolved to overcome the limitations of the anomaly detection model. In this section, we analyze some IDS literature in the last 5 years using KDD Cup 99 and the NSL-KDD dataset, which presents experimental results in the form of detection accuracy for the overall class and detection accuracy for each class.

Lin et al. [2] propose an intrusion detection model using a centroid-based classifier, namely, the cluster center and nearest neighbor (CANN). This approach generates the one-dimensional representative feature from the sum of two distances. A k-NN classifier is used to process the one-dimensional representative feature. The classification accuracy of this approach for the overall class is 99.46%, but the sensitivity of the U2R class is 3.85%, and the sensitivity of the R2L class is 57.05%.

Al-Yaseen et al. [5] propose another approach using SVM and extreme learning machine. They use a modified k-mean to improve the training dataset's quality. This approach can increase the accuracy of the U2R class to 21.93%, but the accuracy of the overall class and the sensitivity of the R2L class is lower than [2]. The overall accuracy is 95.75%, and the sensitivity of the R2L class is 31.39%.

Pajouh et al. [10] propose an IDS model based on a two-tier classifier using naïve Bayes classifier and k-nearest neighbor (k-NN). They use linear discriminant analysis for dimensionality reduction. This approach can produce higher detection accuracy in the U2R and R2L classes than [5], but the overall accuracy is lower than [5]. The overall accuracy is 94.56%, the sensitivity of the U2R class is 67.15%, and the sensitivity of the R2L class is 34.81%.

Thaseen and Kumar [11] propose a multi-class SVM, the Z-score normalization method, and the chi-square feature selection method to recognize the diverse attacks on a network. The experimental results show that their approach produces higher detection accuracy in the U2R and R2L classes than [10]. The overall accuracy is also higher than [10], but still lower than [2]. The overall accuracy is

98.00%, the sensitivity of the U2R class is 73.90%, and the sensitivity of the R2L class is 98.70%.

Bostani and Sheikhan [12] use a modified optimum path forest (OPF) algorithm in intrusion detection. This model can produce higher detection accuracy in the U2R class than [11], but it is lower than [11] in the overall accuracy and the accuracy of the R2L class. The overall accuracy is 91.74%, the sensitivity of the U2R class is 77.98%, and the sensitivity of the R2L class is 81.13%.

Kumar et al. [13] also propose a multi-class SVM to detect intrusion. They use a multi-linear dimensionality reduction (MLDR) method to reduce the dataset's dimensionality. Their experimental results show that this approach also can improve the classification accuracy of SVM. The performance of this model is higher than [12] in the accuracy of the overall class and the U2R class, but for the R2L class, it is lower. The overall accuracy is 98.44%, the sensitivity of the U2R class is 79.77%, and the sensitivity of the R2L class is 78.66%.

Mahendiran and Appusamy [14] propose the other approach using a CRF-based classifier along with a feature selection method using a One-R algorithm for detection intrusion. The performance of this approach is higher than [13] in terms of classification accuracy of the U2R class and the R2L class, but for the overall class, it is lower. The overall accuracy is 98.15%, the sensitivity of the U2R class is 92.30%, and the sensitivity of the R2L class is 96.11%.

In our previous research [9], we propose the intrusion detection model that combines the log normalization, the feature selection method for imbalanced classes, and SVM with parameter optimization. The performance of this approach can outperform the performance of the model in [14] in terms of classification accuracy of the overall class, but for the U2R class and the R2L class, it is lower. The overall accuracy is 99.80%, the sensitivity of the U2R class is 73.08%, and the sensitivity of the R2L class is 93.77%.

In this study, we propose an IDS model that integrates multi-class WSVM with log normalization and the feature selection method for imbalanced classes to improve the accuracy of detecting the minority class and keep capable of detecting another with high accuracy, high sensitivity and high specificity.

## 3. Proposed Work

In this section, we describe our proposed model, which is the integration of multi-class weighted-SVM with an optimization class weight, log normalization, and two feature selection methods. The block diagram of this proposed model is presented in Fig. 1. It has four stages of the process.

We do data preprocessing in the initial stage. In the second stage, we do dimension reduction using two feature selection methods for imbalanced class: the m-RIGFS and the RWIGFS. In the third stage, optimization of the weight of minority class (R2L and U2R) is done by using a grid search method. To help the selection process in the third and fourth stages, we convert several measures into a single score, namely the CPI. In the fourth stage, the Weighted-SVM classifier uses the optimal class weight to train and evaluate the model.

Next, we describe the data pre-processing in subsection 3.1. The Feature Selection methods are described in subsection 3.2. The CPI is described in subsection 3.3. The weighted-SVM classification model is described in subsection 3.4. In subsection 3.5, we describe the one-against-one approach for multiclass SVM.

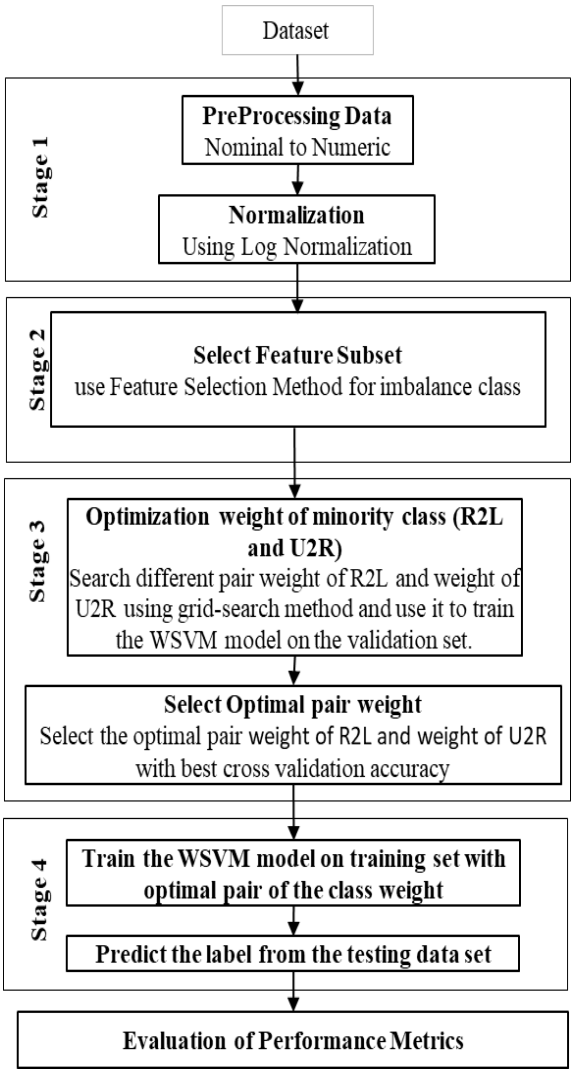

Figure.1 Proposed intrusion detection model

### 3.1 Data pre-processing

Data preprocessing is done by transforming the nominal feature to numeric and then performing the log normalization process on all features.

As in previous studies [9], we implement log normalization in this study. The log normalization scores $x'$ are calculated using Eq. (1), where $x$ is the value of the feature before normalization and $x'$ is its value after normalization.

$$x' = log(1 + x) \tag{1}$$

### 3.2 Feature selection

In this subsection, we describe a original rank information-gain feature selection (RIGFS)[15], a original rank gain-ratio feature selection (RGRFS) [15] and two feature selection methods for imbalanced class (m-RIGFS and RWIGFS) used in this study. We combine the filter-based feature selection and the wrapper-based feature selection approach to get the best feature subset, which supports in detecting the minority classes.

The filter-based approach is used to build a feature ranking based on the value of information-gain features. Information-gain is a standard measurement of the quality of the attributes. In this research, we adopt information-gain and gain-ratio as the attribute evaluation measurement in the proposed feature selection. Information-gain is specified as the amount of information which resulted from the attributes to determine the class, as shown in Eq. (2), where $H$ is the entropy of an outcome or the average amount of information.

$$InfoGain(A) = H_C - H_{C|A} \tag{2}$$

The amount of information from the outcome $X_j$ is defined as a negative logarithmic of its probability, as shown in Eq. (3).

$$I(X_j) = -log_2 P(X_j) \tag{3}$$

If our experiment has $m$ disjoint possible outcomes $X_j$ where $j = 1..m$ and $\sum_j P(X_j) = 1$, then the entropy of outcome $H(X)$ is calculated using Eq. (4).

$$H(X) = -\sum_j^m P(X_j)\, log_2 P(X_j) \tag{4}$$

The gain-ratio is defined as information-gain normalized with the attribute entropy, as shown in Eq. (5).

$$GainRatio(A) = \frac{InfoGain(A)}{H_A} \tag{5}$$

The difference between the RIGFS and m-RIGFS is at the stage of the process of calculating information-gain and ranking. In the RIGFS and RGRFS, the calculation and ranking process is directly carried out on the dataset, while the m-RIGFS is done on the temporary dataset first. In the m-RIGFS method, this process is done on a temporary dataset that only consists of the normal class and 50% of attack classes, which are considered as minority classes, namely the R2L class and U2R class [9]. In the RWIGFS method, we use weighting information-gain from an attribute to sorting the features. This score is obtained by adding up the information-gain attribute value of a class that has been multiplied by the class weight, where the weight of a class is inversely proportional to the amount of data in that class. The information-gain attribute value of each class is counted from the temporary dataset with two classes, such as the first is the class-processed, and the second is the rest.

Next, a wrapper-based approach using SVM is used to select the $n$-best rank feature subset that produces the highest overall accuracy, the lowest false negative, and the highest sensitivity in minority classes (U2R and R2L). To help the selection process, we convert several measures into a single score, namely the CPI. We will choose a subset of features that produce the highest CPI value, where the highest CPI value reflects that the best conditions of overall accuracy, false negative, and sensitivity in minority classes.

### 3.3 Composite performance index

The CPI is generated from four single performance measures (accuracy of all classes, false-negative of attack classes, lowest sensitivity minority class, and lowest second sensitivity minority class). This index is used to get a performance condition that has a high accuracy and all classes detected. We apply this index in the feature selection and the optimization of the weight of minority class in W-SVM.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{6}$$

$$Sensitivity = \frac{TP}{TP+FN} \tag{7}$$

$$Specificity = \frac{TN}{TN+FP} \tag{8}$$

Consider true positive ($TP$) denotes the number of attack samples that are correctly predicted as attacks. The true negative ($TN$) refers to the number of normal network traffic samples correctly classified as normal network traffics. The false positive ($FP$) is defined as the number of normal network traffic samples incorrectly classified as attacks, while a false negative ($FN$) refers to the number of attack samples incorrectly assigned as normal network traffics. Accuracy is the fraction of predictions our model got right; its formula is shown in Eq. (6). It is a measurement of the closeness of the experimental value to the actual amount of the substance in the confusion matrix. Sensitivity or recall is accuracy on the attack samples. Its formula is shown in Eq. (7). On the other hand, specificity is accuracy on the normal network traffics samples. Its formula is shown in Eq. (8).

We apply the MADM problem approach to build CPI. Inspired by Simple Multi-Attribute Rating Technique, one of the multi-criteria decision models discussed in [16], we generate this index. We implement the subjective approach [17] to get the attribute weights according to the decision-makers' subjective option on the set of attributes. Because we require only ranking rather than specifying a precise degree or magnitude, we use the importance of criteria to obtain the attribute weights.

For aggregation purposes, we define a set of non-negative weights $w_j$ to represent the contribution of indicator $I_j$ to the value of the composite indicator index. Indicators are arranged in a sequence of priority in descending order such that $w_1 \geq w_2 \geq \cdots \geq w_J$, where $J$ is the number of indicators and $j$ is the indicator's sequence number. The score of CPI is expressed as a weighted sum of performance measures under multiple indicators. Moreover, the weighted linear model for aggregation purposes of the composite indicators is presented in Eq. (9). The constraints, Eq. (10)-Eq. (13), are some limits for weighting values. Eq. (10) and Eq. (11) are the normalization constraint for the value of weights so that its values are always within the range of 0 and 1. Also, the total number of all weights is equal to 1. Moreover, Eq. (12) is used to ensure a sequence of indicators ranking. The formula in Eq. (13) is used to obtain the weight value based on the order of importance of the indicators [18], which satisfies the constraints in Eqs. (10) – (12).

$$CPI = \sum_{j=1}^{J} w_j . I_j \quad (9)$$

Such that

$$\sum_{j=1}^{J} w_j = 1, \quad (10)$$

$$w_j > 0, \quad j = 1,2,\dots,J \quad (11)$$

$$w_j - w_{(j+1)} \geq 0, \quad j = 1,2,\dots,(J-1) \quad (12)$$

$$w_j = \frac{(J-j+1)}{\sum_{n=1}^{J} n}, \quad j = 1,2,\dots,J \quad (13)$$

We use the accuracy as the first priority, the false-negative as the second priority, and sensitivity for minority classes U2R and R2L in the third and fourth priority. The CPI is calculated using Eq. (14), where $w_1$ is a weight of the accuracy, $w_2$ is a weight of the false negative, $w_3$ is a weight of the first minority class (U2R), and $w_4$ is a weight of the second minority class (R2L).

$$CPI = w_1 \times Accuracy_{Overall} + w_2 \times FalseNegative_{Overall} + w_3 \times Sensitivity_{U2R} + w_4 \times Sensitivity_{R2L} \quad (14)$$

Next, we calculate $w_1, w_2, w_3$, and $w_4$ by using Eq. (13) where $J = 4$.

$$w_1 = \frac{(4-1+1)}{1+2+3+4} = \frac{4}{10} = 0.4$$

We get $w_1$ = 0.4, $w_2$ = 0.3, $w_3$ = 0.2, and $w_4$ = 0.1. So the formula CPI is as follows, as shown in Eq. (15).

$$CPI = 0.4 \times Accuracy_{Overall} + 0.3 \times FalseNegative_{Overall} + 0.2 \times Sensitivity_{U2R} + 0.1 \times Sensitivity_{R2L} \quad (15)$$

### 3.4 Multi-class weighted-support vector machine model

WSVM uses the approach that each data point has a different weight according to its relative importance in class. This approach makes various data points have different contributions to learning decision making.

WSVM constructs a cost function to reduce the classification error and maximize the separation margin. Dissimilar to the penalty term in standard SVM, WSVM weights the penalty term to minimize the effect of less important data points (such as noises and outliers). It assigns the weight $W_i$ to the data point $x_i$. In standard SVM, the value of $C$ is fixed and all training data points are equally treated during the training process.

The support vector technique requires the solution of the optimization problem. Let the

training vector $x_i$ be mapped into a higher-dimensional space by function $\phi$. Slack variable $\xi_i$ is used to measure the deviation of training examples. The parameter $C$ is a user-specified positive parameter, which balances between the solution complexity and the solution error $\sum_{i=1}^{N} W_i \xi_i$. The formula of the optimization problem is as shown in Eq. (16).

$$Minimize\ \Phi(w) = \frac{1}{2} w^T w + C \sum_{i=1}^{N} W_i \xi_i \quad (16)$$

Subject to

$$y_i \left( \left( w, \phi(x_i) \right) + b \right) \geq 1 - \xi_i, \quad i = 1, \ldots, N$$
$$\xi_i \geq 0, \quad i = 1, \ldots, N$$

### 3.5 One-against-one approach

SVM is originally designed only for the classification of two classes. We required to construct a multi-class SVM considering that the NSL-KDD dataset has five class labels. The one-vs-one strategy is a common and established technique in machine learning to deal with multi-class classification problems [19]. In this strategy, a multi-class SVM model with $k$ classes dataset will be constructed from $k(k-1)/2$ SVMs. Each one is trained on data from two classes. A voting procedure was implemented, where the class with the most votes was adopted as an output label of the test data point.

In our research problem that differentiated five types of network traffics, we construct ten WSVMs, and their structural diagram is shown in Fig. 2. The five types of network traffic are Probe, Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R), and Normal.

WSVM-1 for class Normal and class DoS, WSVM-2 for class Normal and class R2L, WSVM-3 for class Normal and class Probe, and WSVM-4 for class Normal and class U2R. WSVM-5 for class DoS and class R2L, WSVM-6 for class DoS and class Probe, and WSVM-7 for class DoS and class U2R. WSVM-8 for class R2L and class Probe, WSVM-9 for class R2L and class Probe, and WSVM-10 for class Probe and class U2R.

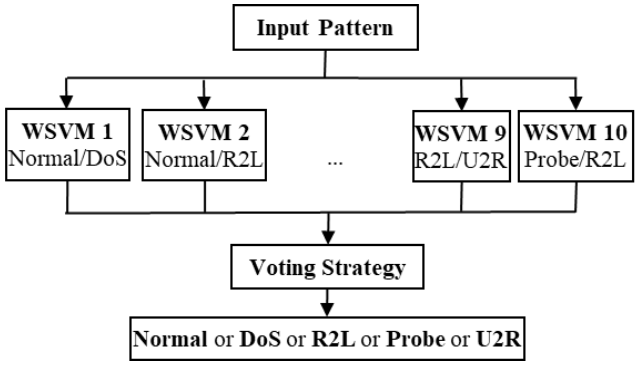


Figure. 2 Structure of multi-class weighted-SVM using OAO approach

The WSVM-1 processes training data composed of class Normal and class DoS, and it also classifies only class Normal and class DoS in testing data. When carrying out the classification process, all SVMs classify the test data, and the results are identified as the class with the highest number of votes.

## 4. Implementation and results

We conduct the experiments by using Java programming and Weka 3.8.3 library [20]. OAO multi-class weighted-SVM with RBF kernel is implemented using the LibSVM [21] package that integrated with Weka. Besides, we perform optimizing the class weight using the grid search method.

We use the NSL-KDD dataset [22], which contains 41 features. This dataset has five classes, namely Normal, Probe, DoS, U2R, and R2L. The experiments use the entire NSL-KDD training dataset, which contains 125,973 records. We use 37791 records (30%) to training data and 88182 records (70%) to the testing set. The training data is taken from 30% of the back of the NSL-KDD training dataset, and the testing set is the rest. Information about the NSL-KDD dataset and its attacks can be found at [23].

The following metrics are used to measure the IDS model's performance: 1)accuracy, 2) sensitivity, 3) specificity, 4) false alarm rate (FAR), and 5) false-negative rate (FNR). Accuracy, sensitivity, and specificity are calculated using Eqs. (6), (7), and (8), respectively. FAR or false-positive rate is formulated as in Eq. (17). On the other hand, FNR or miss rate is formulated as in Eq. (18).

$$FAR = \frac{FP}{FP+TN} = 1 - Specificity \quad (17)$$

$$FNR = \frac{FN}{FN+TP} = 1 - Sensitivity \quad (18)$$

### 4.1 Performance analysis of proposed IDS model

Firstly, we present the implementation of two feature selection methods in the NSL-KDD dataset. We apply m-RIGFS and RWIGFS to get a subset of features whose members are less than 50% of the

Table 2. Top 20 attributes ranking from RWIGFS methods

| Rank | Attribute name and number in KDD dataset | Description |
|---|---|---|
| 1 | src_bytes (5) | Total data bytes from source to destination |
| 2 | service (3) | Network service on the destination, e.g., HTTP, telnet, etc. |
| 3 | dst_bytes (6) | Total data bytes from destination to source |
| 4 | dst_host_srv_count (33) | The number of connections to a destination port |
| 5 | hot (10) | The number of "hot" indicators |
| 6 | count (23) | The number of connections to similar host as the current connection in the past two seconds |
| 7 | dst_host_diff_srv_rate (35) | % of connections that exist for different services between connections in "dst_host_count" |
| 8 | duration (1) | The length of the connection |
| 9 | flag (4) | The status of the connection |
| 10 | dst_host_count (32) | The number of the destination host |
| 11 | diff_srv_rate (30) | % of connections in various services between connections in "count" |
| 12 | srv_count (24) | The number of connections to similar services as the current connection in the past two seconds |
| 13 | dst_host_same_src_port_rate (36) | % of connections to a similar source port |
| 14 | same_srv_rate (29) | % of connections in similar service between connections in "count" |
| 15 | dst_host_serror_rate (38) | % of connections in "dst_host_count" that activated "flag" s3, s2, s1 or s0 |
| 16 | dst_host_srv_diff_host_rate (37) | Different host rate for destination host |
| 17 | dst_host_same_srv_rate (34) | % of connections to a similar service |
| 18 | logged_in (12) | The status of login. If the connection with login's status is true then assign 1 else 0. |
| 19 | root_shell (14) | If a command interpreter with a root account is running root shell the assign 1 else 0 |
| 20 | serror_rate (25) | % of connections in "count" that activated "flag" s3, s2, s1 or s0 |

features of the NSL-KDD dataset, which can produce the best overall accuracy and the best sensitivity in the U2R and R2L classes.

Table 2 shows the sequence of the 20 top features produced in the ranking process by RWIGFS methods. The features ranking from m-RIGFS can look at [18]. Next, we process the 20 top features using the SVM classifier with a wrapper approach to get a subset of features that can produce the best overall accuracy and the best sensitivity in U2R and R2L classes. Previously, we create 20 subset features from the top 20 features.

The first subset contains one top feature, the second subset contains two top features, and so on, where the $n^{th}$ subset contains n top features. We will choose a subset of features that produce the highest CPI value, where the highest CPI value reflects that the best conditions of overall accuracy and sensitivity are minority classes (U2R and R2L).

As shown in Fig. 3, the CPI values from RWIGFS and m-RIGFS are better than original-RIGFS and RGRFS. It represents that the two proposed feature selection methods produce better performance than original-RIGFS and RGRFS. The highest CPI in the m-RIGFS can be achieved by

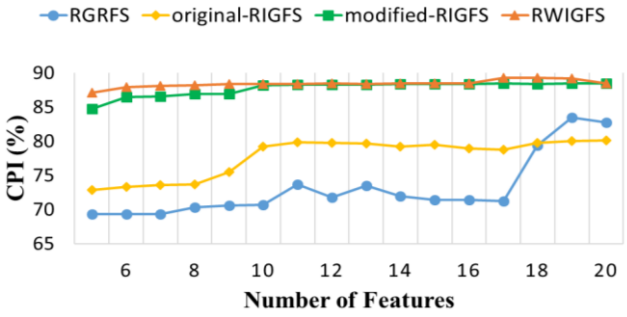


Figure.3 CPI comparison in four feature selection methods

using the top 19 features, whereas the highest CPI in the RWIGFS can be achieved using the top 18 features. Therefore, we chose a subset with 19 features to be used in the intrusion detection model with the m-RIGFS and a subset with 18 features to be used in the intrusion detection model with the RWIGFS.

Next, we apply the selected subset features in WSVM and optimize the minority class weight using the grid search method. Table 3 shows some of the results in optimizing the weights of the minority classes, where c1, c2, c3, c4, and c5 are the Normal, Dos, R2L, Probe, and U2R classes respectively. We test the model on various minority

class weights (R2L class and U2R class), where the range of class weights is from 1 to 10. The majority class weights (Normal, DoS, Probe) are given a value of 1. The aim is to obtain the best accuracy of R2L and U2R while maintaining the majority class accuracy. We apply CPI for ranking the classifier's performance. In m-RIGFS, the selection results show that SVM with a subset of 17 features produces the highest CPI using class weight pairs (Normal, DoS, R2L, Probe, U2R: 1, 1, 3, 1, 10). While in the RWIGFS, SVM with the subset of 19 features produces the highest CPI using the class weight pair (Normal, DoS, R2L, Probe, U2R: 1, 1, 5, 1, 10).

The experimental results show that the proposed IDS model produces relatively high minority class sensitivity and high detection in the overall class. The detection in the U2R attack class is over 56%, and the detection in the R2L attack class is over 92%. Overall accuracy, sensitivity, and specificity reach more than 99%, with false alarms below 0.5% and false-negative rates below 0.7%.

In detecting all classes of attacks, the combination of WSVM with m-RIGFS has higher performance than the combination of WSVM with RWIGFS. But in detecting Normal class and in the overall accuracy, the combination of WSVM with RWIGFS is better. The overall performance of WSVM and m-RIGFS combination is as follows. The overall accuracy is 99.4364%, the overall sensitivity is 99.3957%, the overall specificity is 99.5098%, the false alarm is 0.4902%, and the false-negative rate is 0.6043%. And the following are the sensitivity in each attacks classes (DoS, Probe, R2L, U2R): 99.8757%, 98.3001%, 92.4394%, and 57.50%, respectively.

On the other hand, for a combination of WSVM with RWIGFS, the overall accuracy is 99.4455%, the overall sensitivity is 99.3423%, the overall specificity is 99.5565%, the false alarm is 0.4435%, and the false-negative rate is 0.6577%. And the sensitivity in each attacks classes (DoS, Probe, R2L, U2R) are 99.8198%, 98.2890%, 92.0114%, and 56.4103%, respectively.

This condition can be observed in the confusion matrix in Tables 4 and 5. The columns in the matrix show predicted values, the rows show actual values, and the diagonal entries present the correct prediction.

Next, we compare the performance of the proposed model with the other models. Table 6 shows the performance comparison between the proposed model and the previous IDS models. The performance classification of the proposed model is ranked 3rd and ranked 4th out of 10 models compared in the fields: overall accuracy, specificity, the sensitivity of the DoS class and, the Probe class

Table 3. The accuracy and CPI obtained from the optimizing class weight WSVM with 18 features of the RWIGFS using a grid search

| Class weight (*) | | | | | Overall | Sensitivity | | CPI |
|---|---|---|---|---|---|---|---|---|
| c1 | c2 | c3 | c4 | c5 | Accuracy | U2R | R2L | |
| 1 | 1 | 1 | 1 | 1 | 0.9951 | 0.0000 | 0.9241 | 0.4904 |
| 1 | 1 | 1 | 1 | 2 | 0.9951 | 0.0833 | 0.9241 | 0.5325 |
| 1 | 1 | 1 | 1 | 3 | 0.9951 | 0.2500 | 0.9241 | 0.6196 |
| 1 | 1 | 1 | 1 | 4 | 0.9952 | 0.5833 | 0.9241 | 0.7443 |
| 1 | 1 | 1 | 1 | 5 | 0.9952 | 0.6667 | 0.9241 | 0.7762 |
| 1 | 1 | 1 | 1 | 6 | 0.9951 | 0.6667 | 0.9241 | 0.7762 |
| 1 | 1 | 1 | 1 | 7 | 0.9951 | 0.6667 | 0.9241 | 0.7762 |
| 1 | 1 | 1 | 1 | 8 | 0.9951 | 0.6667 | 0.9241 | 0.7762 |
| 1 | 1 | 1 | 1 | 9 | 0.9952 | 0.6667 | 0.9241 | 0.7762 |
| 1 | 1 | 1 | 1 | 10 | 0.9952 | 0.6667 | 0.9241 | 0.7762 |
| 1 | 1 | 2 | 1 | 10 | 0.9949 | 0.6667 | 0.9276 | 0.7920 |
| **1** | **1** | **3** | **1** | **10** | **0.9948** | **0.6667** | **0.9310** | **0.8082** |
| 1 | 1 | 4 | 1 | 10 | 0.9947 | 0.6667 | 0.9310 | 0.8081 |
| 1 | 1 | 5 | 1 | 10 | 0.9946 | 0.6667 | 0.9310 | 0.8081 |
| 1 | 1 | 6 | 1 | 10 | 0.9945 | 0.5833 | 0.9310 | 0.7914 |
| 1 | 1 | 7 | 1 | 10 | 0.9945 | 0.5833 | 0.9310 | 0.7914 |
| 1 | 1 | 8 | 1 | 10 | 0.9943 | 0.5833 | 0.9310 | 0.7913 |
| 1 | 1 | 9 | 1 | 10 | 0.9943 | 0.5833 | 0.9310 | 0.7913 |
| 1 | 1 | 10 | 1 | 10 | 0.9943 | 0.5833 | 0.9345 | 0.8078 |

***(*)*** *c1=Normal, c2=DoS, c3=R2L, c4=Probe, c5=U2R*

Table 4. Confusion matrix obtained from weighted-SVM [1-1-5-1-10] with 19 features of the m-RIGFS

| | | Prediction | | | | |
|---|---|---|---|---|---|---|
| | | **Normal** | **DoS** | **R2L** | **Probe** | **U2R** |
| Actual | **Normal** | 46891 | 14 | 138 | 67 | 12 |
| | **DoS** | 40 | 32143 | 3 | 4 | 0 |
| | **R2L** | 53 | 0 | 648 | 1 | 3 |
| | **Probe** | 138 | 7 | 0 | 7980 | 0 |
| | **U2R** | 17 | 0 | 0 | 0 | 23 |

Table 5. Confusion matrix obtained from weighted-SVM [1-1-3-1-10] with 18 features of the RWIGFS

| | | Prediction | | | | |
|---|---|---|---|---|---|---|
| | | **Normal** | **DoS** | **R2L** | **Probe** | **U2R** |
| Actual | **Normal** | 46913 | 31 | 95 | 72 | 11 |
| | **DoS** | 58 | 32128 | 0 | 4 | 0 |
| | **R2L** | 56 | 0 | 645 | 1 | 3 |
| | **Probe** | 139 | 1 | 0 | 7985 | 0 |
| | **U2R** | 17 | 0 | 0 | 1 | 22 |

Table 6. Performance comparison of the various IDSs

| Methods | ∑ features | Accuracy (%) | Specificity (%) | Sensitivity (%) | | | | |
|---|---|---|---|---|---|---|---|---|
| | | Overall | Normal | Overall | DoS | R2L | Probe | U2R |
| OneR-FS and CRF [14] | 24 | 98.15 | 98.58 | --- | 98.02 | 96.11 | 96.57 | 92.30 |
| MLDR and multi-class SVM [13] | --- | 98.44 | 95.74 | --- | 95.99 | 78.66 | 94.97 | 79.77 |
| Modified OPF [12] | --- | 91.74 | 98.55 | --- | 96.89 | 81.13 | 85.92 | 77.98 |
| Chi-FS and multi-class SVM [11] | 31 | 98.00 | 99.60 | --- | 99.90 | 98.70 | 99.20 | 73.90 |
| SVM-OP+log normalization [9] | 17 | 99.80 | 99.84 | --- | 99.96 | 93.77 | 99.61 | 73.08 |
| Naïve Bayes and CF-KNN [10] | --- | 94.56 | 94.56 | --- | 84.68 | 34.81 | 79.76 | 67.16 |
| Hybrid SVM and ELM [5] | --- | 95.75 | 98.13 | --- | 99.54 | 31.39 | 87.22 | 21.93 |
| CANN [2] | 19 | 99.46 | 97.04 | --- | 99.68 | 57.05 | 87.61 | 3.85 |
| Proposed Approach | | | | | | | | |
| WSVM and m-RIGFS | 19 | 99.44 | 99.51 | 99.40 | 99.85 | 92.44 | 98.30 | 57.50 |
| WSVM and RWIGFS | 18 | 99.45 | 99.56 | 99.34 | 99.82 | 92.01 | 98.29 | 56.41 |

In the sensitivity of the R2L class, the proposed model is ranked 4th and ranked 5th, and in the sensitivity of the U2R class is ranked 7th and ranked 8th. However, those models that rank higher in minority classes use more features and also use fewer data samples for training and testing.

### 4.2 Discussions

The proposed model is a combination of multi-class weighted-SVM optimized by tuning weighted class techniques, and two feature selection methods for imbalanced classes (m-RIGFS and RWIGFS). These methods are dissimilar with the approach commonly used to avoid high dimensional curses in large data sets. The m-RIGFS method and RWIGFS are created to produce a subset of features that support better detection in minority classes. Multi-class weighted-SVM in one-against-one mode is implemented to get high detection accuracy in all classes. Furthermore, the weighted minority class optimization of the SVM model is carried out to produce better predictions. The novelty of this approach is the use of a CPI in the feature selection and the optimization of the weight class in W-SVM.

## 5. Conclusions

The intrusion detection model proposed in this study a multi-class weighted-SVM and two feature selection methods for imbalanced classes (m-RIGFS and RWIGFS). The experimental results show that the proposed IDS model can produce overall accuracy, sensitivity, and specificity that reaches more than 99%, with false alarms below 0.5% and false-negative rates below 0.7%, but the sensitivity of the U2R class is still below 60%.

For future enhancements, we want to develop an ensemble IDS using weighted-SVM and SVM with a parameter optimization technique and implement to others IDS dataset.

## References

[1] N. B. Abdel-Hamid, S. ElGhamrawy, A. ElDesouky, and H. Arafat, "A Dynamic Spark-based Classification Framework for Imbalanced Big Data", *Journal of Grid Computing*, Vol. 16, No. 4, pp. 607–626, 2018.

[2] W. C. Lin, S. W. Ke, and C. F. Tsai, "CANN: An intrusion detection system based on combining cluster centers and nearest neighbors", *Knowledge-Based Systems*, Vol. 78, No. 1, pp. 13–21, 2015.

[3] T. Ahmad and K. Muchammad, "L-SCANN: Logarithmic subcentroid and nearest neighbor", *Journal of Telecommunications and Information Technology*, Vol. 2016, No. 4, pp. 71–80, 2016.

[4] I. Z. Muttaqien and T. Ahmad, "Increasing performance of IDS by selecting and transforming features" In: *Proc. of 2016 IEEE International Conference on Communication, Network, and Satellite*, pp. 85–90, 2017.

[5] W. L. Al-Yaseen, Z. A. Othman, and M. Z. A. Nazri, "Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system", *Expert Systems with Applications*, Vol. 67, pp. 296–303, 2017.

[6] Y. Zhao, Z. S.-Y. Wong, and K. L. Tsui, "A Framework of Rebalancing Imbalanced Healthcare Data for Rare Events' Classification: A Case of Look-Alike Sound-Alike Mix-Up Incident Detection", *Journal of Healthcare Engineering*, Vol. 2018, No. 2010, pp. 1–11, 2018.

[7] X. Yang, Q. Song, and Y. Wang, "A weighted support vector machine for data classification", *International Journal Pattern Recognition and Artificial Intelligence*, Vol. 21, No. 5, pp. 961–976, 2007.

[8] A. A. Aburomman and M. B. I. Reaz, "A novel weighted support vector machines multiclass classifier based on differential evolution for intrusion detection systems", *Information Science (Ny).*, Vol. 414, pp. 225–246, 2017.

[9] B. Setiawan, S. Djanali, and T. Ahmad, "Increasing accuracy and completeness of intrusion detection model using fusion of normalization, feature selection method and support vector machine", *International Journal Intelligent Engineering and Systems*, Vol. 12, No. 4, pp. 378–389, 2019.

[10] H. H. Pajouh, G. H. Dastghaibyfard, and S. Hashemi, "Two-tier network anomaly detection model: a machine learning approach", *Journal of Intelligence Information Systems*, Vol. 48, No. 1, pp. 61–74, 2017.

[11] I. S. Thaseen and C. A. Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM", *Journal of King Saud University-Computer and Information Sciences*, Vol. 29, No. 4, pp. 462–472, 2017.

[12] H. Bostani and M. Sheikhan, "Modification of supervised OPF-based intrusion detection systems using unsupervised learning and social network concept", *Pattern Recognition*, Vol. 62, pp. 56–72, 2017.

[13] B. N. Kumar, M. S. V. S. B. Raju, and B. V. Vardhan, "Enhancing the performance of an intrusion detection system through multi- linear dimensionality reduction and Multi-class SVM", *International Journal of Intelligent Engineering Systems*, Vol. 11, No. 1, pp. 181–192, 2018.

[14] A. Mahendiran and R. Appusamy, "An intrusion detection system for network security situational awareness using conditional random fields", *International Journal of Intelligent Engineering Systems*, Vol. 11, No. 3, pp. 196–204, 2018.

[15] J. Novaković, P. Strbac, and D. Bulatović, "Toward optimal feature selection using ranking methods and classification algorithms", *Yugoslav Journal of Operations Research*, Vol. 21, No. 1, pp. 119–135, 2011.

[16] A. Ishizaka and S. Siraj, "Are multi-criteria decision-making tools useful? An experimental comparative study of three methods", *European Journal of Operational Research*, Vol. 264, No. 2, pp. 462–471, 2018.

[17] Y. Dong, Y. Liu, H. Liang, F. Chiclana, and E. Herrera-Viedma, "Strategic weight manipulation in multiple attribute decision making", *Omega (United Kingdom)*, Vol. 75, pp. 1339–1351, 2018.

[18] B. Setiawan, S. Djanali, and T. Ahmad, "Assessing Centroid-Based Classification Models for Intrusion Detection System Using Composite Indicators", *Procedia Computer Science*, Vol. 161, pp. 665-676, 2019.

[19] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "DRCW-OVO: Distance-based relative competence weighting combination for One-vs-One strategy in multi-class problems", *Pattern Recognition*, Vol. 48, No. 1, pp. 28–42, 2015.

[20] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, Fourth Edition. Morgan Kaufmann, 2016.

[21] C. Chang and C. Lin, "LIBSVM : A Library for Support Vector Machines", *ACM Transactions on Intelligent Systems Technology*, Vol. 2, No. 3, p. 27, 2011.

[22] Canadian Institute of Cybersecurity, "NSL-KDD dataset," 2009. Accessed on: December 3, 2019. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html.

[23] D. Protić, "Review of KDD Cup '99, NSL-KDD and Kyoto 2006+ datasets", *Vojnotehnicki Glasnik*, Vol. 66, No. 3, pp. 580–596, 2018.