IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

IJARSCT
Impact Factor: 6.252

Volume 2, Issue 6, June 2022

# Intrusion Detection System Using K-Means and Edited Nearest Neighbour Algorithm.

**Mr. Abdul Khadar A[1], Modem Tharun Kumar[2], Sharath K N[3], Sukesh V N[4], Tejaswini K N[5]**

Assistant Professor, Department of Information Science and Engineering[1]
Students, Department of Information Science and Engineering[2,3,4,5]
S.J.C. Institute of Technology, Chikkaballapur, Karnataka, India

**Abstract:** *In imbalanced network traffic, malicious cyber-attacks can often hide in large amounts of normal data. It exhibits a high degree of stealth and obfuscation in cyberspace, making it difficult for Network Intrusion Detection System (NIDS) to ensure the accuracy and timeliness of detection. This paper researches machine learning and deep learning for intrusion detection in imbalanced network traffic. It proposes a novel Difficult Set Sampling Technique (DSSTE) algorithm to tackle the class imbalance problem. First, use the Edited Nearest Neighbor (ENN) algorithm to divide the imbalanced training set into the difficult set and the easy set. Next, use the K- Means algorithm to compress the majority samples in the difficult set to reduce the majority. Zoom in and out the minority samples' continuous attributes in the difficult set synthesize new samples to increase the minority number. Finally, the easy set, the compressed set of majority in the difficult, and the minority in the difficult set are combined with its augmentation samples to make up a new training set. The algorithm reduces the imbalance of the original training set and provides targeted data augment for the minority class that needs to learn. It enables the classifier to learn the differences in the training stage better and improve classification performance. To verify the proposed method, we conduct experiments on the classic intrusion dataset NSL-KDD. We use classical classification models: random forest(RF), Support Vector Machine (SVM), XGBoost, Long and Short- term Memory (LSTM), Adaboost, AlexNet, Mini- VGGNet.*

**Keywords:** IDS, Imbalanced Network traffic, Machine Learning, Deep Learning

## I. INTRODUCTION

In the field of machine learning, the problem of category imbalance has always been a challenge. Therefore, intrusion detection also faces enormous challenges in network traffic with extremely imbalanced categories. Therefore, many scholars have begun to study how to improve the intrusion recognition accuracy of imbalanced network traffic data. Piyasak proposed a method to improve the accuracy of minority classification. This method combines the Synthetic Minority Over-sampling Technique (SMOTE) and Complementary Neural Network(CMTNN) to solve imbalanced data classification. Experiments on the UCI dataset show that the proposed combination technique can improve class imbalance problems. Yan proposed an improved local adaptive composite minority sampling algorithm (LA-SMOTE) to deal with the network traffic imbalance problem and then based on the deep learning GRU neural network to detect the network traffic anomaly. Abdul hammed et al. deal with the imbalanced dataset using data Up sampling and Down sampling methods, and by Deep Neural Networks, Random Forest, Voting, Variational Autoencoder, and Stacking Machine Learning classifiers to evaluate datasets. In their proposed method, the accuracy can reach 99.99%. In imbalanced network traffic, different traffic data types have similar representations, especially minority attacks can hide among a large amount of normal traffic, making it difficult for the classifier to learn the differences between them during the training process. In the similar samples of the imbalanced training set, the majority class is redundant noise data. The number is much larger than the minority class, making the classifier unable to learn the distribution of the minority class, so we compress the majority class. The minority class discrete attributes remain constant, and there are differences in continuous attributes. Therefore, the minority class's continuous attributes are zoomed to produce data that conforms to the true distribution. Therefore, we propose the DSSTE algorithm to reduce the imbalance.

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training

data", in order to make predictions or decisions without being explicitly programmed to perform the task. Machine learning algorithms are used in a wide variety of applications, such as email filtering and computer vision, where it is difficult or infeasible to develop a conventional algorithm for effectively performing the task. Machine learning is closely related to computational statistics, which focuses on making predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a field of study within machine learning, and focuses on exploratory data analysis through unsupervised learning. In its application across business problems, machine learning is also referred to as predictive analytics.
Several security solutions have been proposed to detect network abnormal behavior. However, successful attacks is still a big concern in computer society. Lots of security breaches, like Distributed Denial of Service (DDoS), botnets, spam, phishing, and so on, are reported every day, while the number of attacks are still increasing. To overcome this problem, we are planning to implement this project.

## II. RELATED WORK

Before building our application, the following system is taken into consideration:
Title: Toward credible evaluation of anomaly- based intrusion-detection methods
**Authors: M. Tavallaee, N. Stakhanova, and A. A. Ghorbani**
**Abstract:** Since the first introduction of anomaly- based intrusion detection to the research community in 1987, the field has grown tremendously. A variety of methods and techniques introducing new capabilities in detecting novel attacks were developed. Most of these techniques report a high detection rate of 98% at the low false alarm rate of 1%. In spite of the anomaly- based approach's appeal, the industry generally favors signature-based detection for mainstream implementation of intrusion-detection systems. While a variety of anomaly-detection techniques have been proposed, adequate comparison of these methods' strengths and limitations that can lead to potential commercial application is difficult. Since the validity of experimental research in academic computer science, in general, is questionable, it is plausible to assume that research in anomaly detection shares the above problem. The concerns about the validity of these methods may partially explain why anomaly- based intrusion-detection methods are not adopted by industry. To investigate this issue, we review the current state of the experimental practice in the area of anomaly-based intrusion detection and survey 276 studies in this area published during the period of 2000-2008. We summarize our observations and identify the common pitfalls among surveyed works.

**Advantages:**
It is possible that many of the identified pitfalls were avoided in the conducted research, but not reported.

**Disadvantages:**
We can lose its value behind an ambiguous, unclear and unsound presentation.

Title: A macro-social exploratory analysis of the rate of interstate cyber-victimization
**Authors: H. Song, M. J. Lynch, and J. K. Cochran Abstract:** This study examines whether macro-level opportunity indicators affect cyber-theft victimization. Based on the arguments from criminal opportunity theory, exposure to risk is measured by state-level patterns of internet access (where users access the internet). Other structural characteristics of states were measured to determine if variation in social structure impacted cyber-victimization across states. The current study found that structural conditions such as unemployment and non-urban population are associated with where users access the internet. Also, this study found that the proportion of users who access the internet only at home was positively associated with state-level counts of cyber-theft victimization. The theoretical implications of these findings are discussed.
**Advantages:**
We examine effects of macro-social opportunity factors on state-level cyber-theft victimization. Based on theoretical arguments and prior research findings related to COT and cybercrime victimization, we hypothesized that online routine activities related to where users access the internet would affect cyber-theft victimization.

DOI: 10.48175/IJARSCT-5052

**Disadvantages**:

Sample size in the current study could weaken statistical power, which is the probability of rejecting a false null hypothesis, and lead to the insignificance of these two online routine activities.

Title: Incremental anomaly-based intrusion detection system using limited labeled data
Authors: P. Alaei and F. Noorbehbahani
**Abstract:** With the proliferation of the internet and increased global access to online media, cybercrime is also occurring at an increasing rate. Currently, both personal users and companies are vulnerable to cybercrime. A number of tools including firewalls and Intrusion Detection Systems (IDS) can be used as defense mechanisms. A firewall acts as a checkpoint which allows packets to pass through according to predetermined conditions. In extreme cases, it may even disconnect all network traffic. An IDS, on the other hand, automates the monitoring process in computer networks. The streaming nature of data in computer networks poses a significant challenge in building IDS. In this paper, a method is proposed to overcome this problem by performing online classification on datasets. In doing so, an incremental naive Bayesian classifier is employed. Furthermore, active learning enables solving the problem using a small set of labeled data points which are often very expensive to acquire. The proposed method includes two groups of actions i.e. offline and online. The former involves data preprocessing while the latter introduces the NADAL online method. The proposed method is compared to the incremental naive Bayesian classifier using the NSL-KDD standard dataset. There are three advantages with the proposed method: (1) overcoming the streaming data challenge; (2) reducing the high cost associated with instance labeling; and (3) improved accuracy and Kappa compared to the incremental naive Bayesian approach. Thus, the method is well-suited to IDS applications.

**Advantages:**

Improve efficiency in classifying data in an online fashion.
Active learning was used to reduce labeling costs.

**Disadvantages:**

Improving classification accuracy in data with class imbalance so that the data are equally distributed among the training classes.

Using online feature extraction methods in the NADAL framework.
Title: Modeling and implementation approach to evaluate the intrusion detection system
**Authors: M. Saber, S. Chadli, M. Emharraf, and I. El Farissi**
**Abstract:** Intrusions detection systems (IDSs) are systems that try to detect attacks as they occur or when they were over. Research in this area had two objectives: first, reducing the impact of attacks; and secondly the evaluation of the system IDS. Indeed, in one hand the IDSs collect network traffic information from some sources present in the network or the computer system and then use these data to enhance the systems safety. In the other hand, the evaluation of IDS is a critical task. In fact, its important to note the difference between evaluating the effectiveness of an entire system and evaluating the characteristics of the system components. In this paper, we present an approach for IDS evaluating based on measuring the performance of its components. First of all, in order to implement the IDS SNORT components safely we have proposed a hardware platform based on embedded systems. Then we have tested it by using a generator of traffics and attacks based on Linux KALI (Backtrack) and Metasploite 3 Framework. The obtained results show that the IDS performance is closely related to the characteristics of these components.

**Advantages:**

It reduces the impact of attacks; The evaluation of the system IDS.

**Disadvantages:**

It takes more computational time. Its cannot classify the sub attacks.

DOI: 10.48175/IJARSCT-5052

## III. METHODOLOGY USED

The below figure illustrates the steps in intrusion detection system, Data pre-processing first performed in our intrusion detection structure, including duplicate, outlier, and missing value processing. Then, partitioning the test set and the training set, and the training set processed for data balancing using our proposed DSSTE algorithm. Before modeling, to increase the speed of the convergence, we use Standard Scaler to standardize the data and digitize the sample labels. Finally, the processed training set is used to train the classification model, and then the model is evaluated by the test set.

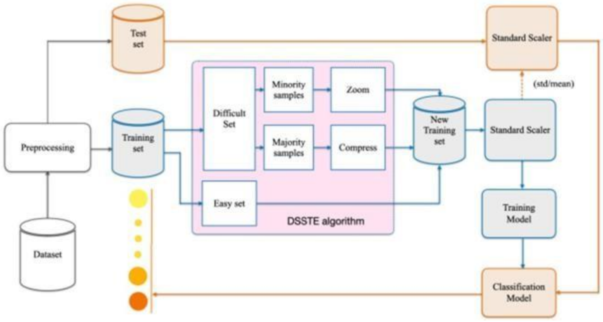

Figure 1 : The Overall Framework Of Intrustion Detection System

### DSSTE Algorithm

Difficult Set Sampling Technique (DSSTE) improve the performance of imbalanced network intrusion detection based on machine learning and deep learning. First, the imbalanced training set to divide into near-neighbor set and far-neighbor set by Edited Nearest Neighbor (ENN) algorithm. The samples in the near-neighbor set are highly similar, making it very difficult for the classifier to learn the differences between the categories, so we refer to the samples in the near- neighbor set as difficult samples and the far-neighbor set as easy samples. Next, we zoom in and out the minority samples in difficult set.

### Algorithm

**Input:** Imbalanced training set S, scaling factor K

**Output:** New training set $S_N$

```
Step 1: Distinguish easy set and difficult set
Take all samples from S and set it as $S_E$
    foreach sample ∈ $S_E$ do
        Compute its K nearestneighbors
        Remove whose most K nearest neighbor samples are of different classes from $S_E$
end for
Easy set $S_E$ , difficultset $S_D$ S=$S_E$
Step 2 : Compress the majority samples in difficult set by the cluster centroid
Take all the majority samples from $S_D$ and set it as $S_{Maj}$
Use K-Means algorithm with K cluster
Use the coordinates of the K cluster centroids replace the majority samples in $S_{Maj}$
Compressed the majority samples set $S_{Maj}$
Step 3 : Zoom augmentation
Take the minority samples from $S_D$ and set it as $S_{Min}$
Take the Discrete attributes from $S_{Min}$ and set it as $X_D$
Take the Continuous attributes from $S_{Min}$ and set it as $X_C$
Take the Label attributes from $S_{Min}$ and set it as Y
for n range(K,K+ $\frac{\text{number}}{}$) do $S_{Min}$,shape[0]
    ∈ XD1 = XD
        XC1 = XC × (1 − 1/n )
        XD2 = XD
```

DOI: 10.48175/IJARSCT-5052

```
        $X_{C2} = X_C \times (1 + 1/n)$
        $S_Z$ append [concat($X_{D1}$, $X_{C1}$, Y), concat($X_{D2}$, $X_{C2}$, Y)]
end for
New training set $S_N = S_E + S_{Maj} + S_{Min} + S_Z$
```

**Random Forest :** Random Forest is an excellent supervised learning algorithm that can train a model to predict which classification results in a certain sample type belong to based on a given dataset's characteristic attributes and classification results. Random Forest is based on a decision tree and adopts the Bagging(Bootstrap aggregating) method to create different training sample sets. The random subspace division strategy selects the best attribute from some randomly selected attributes to split internal nodes.

**Support Vector Machine:** It shows many unique advantages in a small sample, nonlinear, and high- dimensional pattern recognition and can be extended to other functions such as function fitting Machine learning problems. Before the rise of deep learning, SVM was considered the most successful and best- performing machine learning method in recent decades. The SVM method is based on the Vapnik Chervonenkis(VC) dimension theory of statistical learning theory and the principle of structural risk minimization. Its basic idea is to find a separation hyperplane between different categories, so that different category can be better separated.

**XGBoost:** XGBoost is a parallel regression tree model that combines the idea of Boosting, which is improved based on gradient descent decision tree by Chen and Guestrin. Compared with the GBDT(Gradient Boosting Decision Tree) model, XGBoost overcomes the limited calculation speed and accuracy. XGBoost adds regularization to the original GBDT loss function to prevent the model from overfitting. The traditional GBDT performs a first- order Taylor expansion on the calculated loss function and takes the negative gradient value as the residual value of the current model.

**Long Short-Term Memory:** The Long Short-Term Memory(LSTM) Like most RNN, the LSTM network is universal because as long as there is a suitable weight matrix, the LSTM network can calculate any network element that can be calculated by any conventional computer. Different from the traditional RNN, the LSTM network is very suitable for learning from experience. When there is a time lag of unknown size and boundary between important events, the time series can be classified, processed, and predicted. LSTM is not sensitive to gap length and has advantages over other RNN and hidden Markov models and other sequence learning methods in many applications. Here we use K-Means and Edited Nearest Neighbour algorithm for better detection of intrusion.

**AlexNet :** AlexNet is one of the classic basic networks of deep learning. It was proposed by Hinton and his student Alex Krizhevsky in 2012. Its main structure is an 8-layer deep neural network, including 5-layer convolutional layers and 3-layer fully connected layers, which are not counted in the Activation layer and pooling layer. The ReLU function is used as the activation function in the AlexNet convolutional layer, instead of the Sigmoid function widely used in previous networks. Here we use K- Means and Edited Nearest Neighbour algorithm for better detection of intrusion.

**MINI-VGGNet :**The main contribution is to use a small convolution kernel (3 × 33 × 3) to construct various depths of convolutional neural network structures. Moreover, it evaluated these network structures and finally proved that the 16-19 layer network depth could achieve better recognition accuracy. VGG-16 and VGG-19 are commonly used to extract image features. VGG can be regarded as a deepened version of AlexNet. The entire network is superimposed by a convolutional layer and a fully connected layer. Unlike AlexNet, VGGNet uses a small-sized convolution kernel(3 × 3).

**ADABoost :** AdaBoost also called Adaptive Boosting is a technique in Machine Learning used as an Ensemble Method. The most common algorithm used with AdaBoost is decision trees with one level that means with Decision trees with only 1 split. These trees are also called Decision Stumps. What this algorithm does is that it builds a model and gives equal weights to all the data points. It then assigns higher weights to points that are wrongly classified. Now all the points which have higher weights are given more importance in the next model. It will keep training models until and unless a lower error is received.

## IV. IMPLEMENTATION

**Collecting Dataset :** NSL-KDD is the most classic dataset in the field of intrusion detection. It is an improvement based on the KDD99 dataset and is reasonably divided into different difficulty levels in the test set. Although it still has some problems and is not a perfect representation of the existing real network, it can still be used as an effective benchmark dataset to help researchers compare different intrusion detection methods. Each sample in NSL-KDD includes 41 features listed below.

| Attributes | Description |
|---|---|
| 1-9 | Basic features of network connections |
| 10-22 | Content-related traffic features |
| 23-31 | Time-related traffic features |
| 32-41 | Host-based traffic features |

**Table 1:** Dataset Description

**Data Pre-Processing :** When the dataset is extracted, part of the data contains some noisy data, duplicate values, missing values, infinity values, etc. due to extraction errors or input errors. Therefore, we first perform data pre-processing. The main work is as follows.

Duplicate values: delete the sample's duplicate value, only keep one valid data.

Outliers: in the sample data, the sample size of missing values(Not a Number, NaN) and Infinite values(Inf) is small, so we delete this.

Pseudocode :

```
Step 1 : Read the Dataset
Step 2 : for all records in dataset
// Find nan values  If any nan values
Drop the values
Else
// Find repeated values If any duplicate
Drop the value Return pre-processed dataset.
```

**Building Models :** After Pre-processing we are building the intrusion detection system model using the following algorithms.

1. Random Forest,
2. SVM,
3. XGBoost,
4. Adaboost,
5. LSTM (Using K-Means and ENN Algorithm),
6. AlexNet (UsingK-Means and ENN Algorithm),
7. Mini-VGGNet.

Pseudocode:

```
Step 1: Importing library from Sklearn and Tensorflow
Step 2: Build Random Forest, SVM, XGBoost, LSTM and AlexNet (Using K-Means and ENN Algorithm), Mini-VGGNet.
Step 3: Split preprocessed dataset into train set and test set
Step 4: Train the modules using fit() Step 5: Save training model
```

**Performance Analaysis :** We use the Accuracy, Prediction, Recall, and F1-Score to evaluate the experimental model's performance. These evaluation criteria reflect the performance of the intrusion detection system's flow recognition accuracy rate, and false alarm rate. The combination of the model prediction results and the true label is divided into four types:

False Negative(FN) : a positive sample, which is mistakenly judged as a negative sample.

False Positive(FP) : negative samples are misjudged as positive samples.

True Negative(TN) : actually negative samples, are correctly judged as negative samples.

True Positive(TP) : actually positive samples, are judged as the positive sample.

DOI: 10.48175/IJARSCT-5052

Pseudocode:

```
Step 1: Load the test set
Step 2: Load the pre-train model
Step 3: Evaluate the model using predict() Step 4: Get the accuracy
Step 5: Show performance in the graph
```

## V. RESULTS

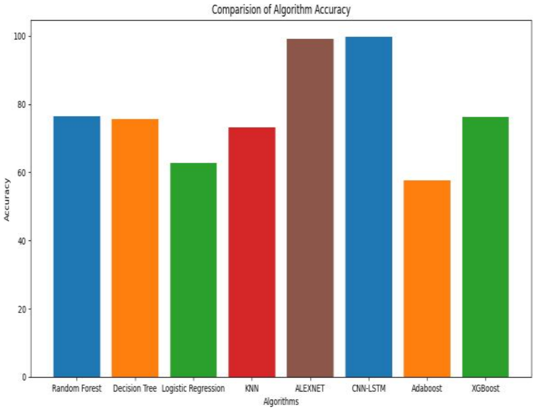

Figure 2 : Comparison of Algorithms accuracy.

## VI. CONCLUSION

As network intrusion continues to evolve, the pressure on network intrusion detection is also increasing. In particular, the problems caused by imbalanced network traffic make it difficult for intrusion detection systems to predict the distribution of malicious attacks, making cyberspace security face a considerable threat. This paper proposed a novel Difficult Set Sampling Technique(DSSTE) algorithm, which enables the classification model to strengthen imbalanced network data learning. A targeted increase in the number of minority samples that need to be learned can reduce the imbalance of network traffic and strengthen the minority's learning under challenging samples to improve the classification accuracy. We used six classical classification methods in machine learning and deep learning and combined them with other sampling techniques. Experiments show that our method can accurately determine the samples that need to be expanded in the imbalanced network traffic and improve the attack recognition more effectively.

## ACKNOWLEDGMENT

The authors gratefully acknowledge the contributions of Ass. Professor Abdul Khadar A and authors who made their contributions to the earlier part of the work which had been previously published.

## REFERENCES

**[1].** D. E. Denning, ''An intrusion-detection model,'' IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222– 232, Feb. 1987.

**[2].** N. B. Amor, S. Benferhat, and Z. Elouedi, ''Naive Bayes vs decision trees in intrusion detection systems,'' in Proc. ACM Symp. Appl. Comput. (SAC), 2004, pp. 420–424.

**[3].** M. Panda and M. R. Patra, ''Network intrusion detection using Naive Bayes,'' Int. J. Comput. Sci. Netw. Secur., vol. 7, no. 12, pp. 258–263, 2007.

**[4].** M. A. M. Hasan, M. Nasser, B. Pal, and S. Ahmad, ''Support vector machine and random forest modeling for intrusion detection system (IDS),'' J. Intell. Learn. Syst. Appl., vol. 6, no. 1, pp. 45–52, 2014.

**[5].** N. Japkowicz, ''The class imbalance problem: Significance and strategies,'' in Proc. Int. Conf. Artif. Intell., vol. 56, 2000, pp. 111–117.

**[6].** Y. LeCun, Y. Bengio, and G. Hinton, ''Deep learning,'' Nature, vol. 521, no. 7553, pp. 436–444, 2015.

**[7].** Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and

DOI: 10.48175/IJARSCT-5052

**[8].** M. S. Lew, ‘‘Deep learning for visual understanding: A review,’’ Neurocomputing, vol. 187, pp. 27–48, Apr. 2016.

**[9].** T. Young, D. Hazarika, S. Poria, and E. Cambria, ‘‘Recent trends in deep learning based natural language processing [review article],’’ IEEE Comput. Intell. Mag., vol. 13, no. 3, pp. 55–75, Aug. 2018.

**[10].** N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, ‘‘A deep learning approach to network intrusion detection,’’ IEEE Trans. Emerg. Topics Comput. Intell., vol. 2, no. 1, pp. 41–50, Feb. 2018.

**[11].** D. A. Cieslak, N. V. Chawla, and A. Striegel, ‘‘Combating imbalance in network intrusion datasets,’’ in Proc. IEEE Int. Conf. Granular Comput., May 2006, pp. 732–737.

**[12].** M. Zamani and M. Movahedi, ‘‘Machine learning techniques for intrusion detection,’’ 2013, arXiv:1312.2177. [Online]. Available: http://arxiv. org/abs/1312.2177

**[13].** M. S. Pervez and D. M. Farid, ‘‘Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs,’’ in Proc. 8th Int. Conf. Softw., Knowl., Inf. Manage. Appl. (SKIMA), Dec. 2014,pp.1–6

DOI: 10.48175/IJARSCT-5052