Global Journal of Computer Science and Technology: E
Network, Web & Security
Volume 20 Issue 2 Version 1.0 Year 2020
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

Anomaly Intrusion Detection based on Concept Drift

By Pradheep D, Gokul R, Naveen V & Vijayarani J
Anna University

Abstract- Nowadays, security on the internet is a vital issue and therefore, intrusion detection is one of the major research problems for networks that defend external attacks. Intrusion detection is a new approach for providing security in existing computers and data networks. An Intrusion Detection System is a software application that monitors the system for malicious activities and unauthorized access to the system. An easy accessibility condition causes computer networks vulnerable against the attack and several threats from attackers. Intrusion Detection System is used to analyze a network of interconnected systems for avoiding uncommon intrusion or chaos. The intrusion detection problem is becoming a challenging task due to the increase in computer networks since the increased connectivity of computer systems gives access to all and makes it easier for hackers to avoid their traces and identification. The goal of intrusion detection is to identify unauthorized use, misuse and abuse of computer systems. This project focuses on algorithms: (i) Concept Drift based ensemble Incremental Learning approach for anomaly intrusion detection, and (ii) Diversity and Transfer-based Ensemble Learning. These are highly ranked anomaly detection models. We study and compare both learning models. The Network Security Laboratory-Knowledge Discovery and Data Mining (NSL-KDD99) dataset have been used for training and to detect the misuse activities.

GJCST-E Classification: *J.7*

Strictly as per the compliance and regulations of:

Anomaly Intrusion Detection based on Concept Drift

Pradheep D[α], Gokul R[σ], Naveen V[ρ] & Vijayarani J[ω]

Abstract- Nowadays, security on the internet is a vital issue and therefore, intrusion detection is one of the major research problems for networks that defend external attacks. Intrusion detection is a new approach for providing security in existing computers and data networks. An Intrusion Detection System is a software application that monitors the system for malicious activities and unauthorized access to the system. An easy accessibility condition causes computer networks vulnerable against the attack and several threats from attackers. Intrusion Detection System is used to analyze a network of interconnected systems for avoiding uncommon intrusion or chaos. The intrusion detection problem is becoming a challenging task due to the increase in computer networks since the increased connectivity of computer systems gives access to all and makes it easier for hackers to avoid their traces and identification. The goal of intrusion detection is to identify unauthorized use, misuse and abuse of computer systems. This project focuses on algorithms: (i) Concept Drift based ensemble Incremental Learning approach for anomaly intrusion detection, and (ii) Diversity and Transfer-based Ensemble Learning. These are highly ranked anomaly detection models. We study and compare both learning models. The Network Security Laboratory-Knowledge Discovery and Data Mining (NSL-KDD99) dataset have been used for training and to detect the misuse activities.

I. Introduction

The Internet has a number of challenges to make it a secure system as it has a large amount of data and information. Computer networks are widely used by businesses, industries and various fields of day to day activity. Advances in technology and business, forced organizations and institutions worldwide to invent and use modern networks for safety. There are many types of attacks threatening computer networks. Dynamic mechanisms can be exploited though security can be ensured through the installation of firewalls and defending software. An intrusion detection system is one of the dynamic mechanisms that determines the specific goal of detecting attacks. An intrusion detection system (IDS) is one of the implemented solutions against hackers and attackers. Moreover, attackers always keep changing their techniques and tools for hacking the network. Intrusion detection system monitors the network and analyzes them to detect any kind of abnormalities, which are harmfull to computer security. There are two methods of intrusion detection (i) Misuse (ii) Anomaly.

A misuse detection system uses recognized patterns for detection, which is also called signature-based detection. A key benefit of these systems is that the patterns or signatures can easily develop and understand the network behavior, if familiar. Misuse aims to determine the attack signatures in the monitored resource. This technique is effective at detecting attacks that are already known. The time taken to match with the patterns stored in the database is minimal. Anomaly detection systems rely on constructing a model of user behavior that is considered normal. The detection of novel attacks is more successful using the anomaly detection approach for an intrusion. This is achieved by using machine learning methods to examine network traffic. Anomaly depends on knowledge of normal behavior and any deviation from normal behavior. Anomaly detection has gained popularity as it became effective against new anomaly attacks.

Concept drift is a change in the characteristics of the data stream. It means that the characteristics of the decision attributes and of the classes to be predicted, change in time in an unpredictable manner is called as concept drift. Such a situation may cause a decrease in classification quality and degrade learning mechanisms. Concept drift in machine learning refers to the change in the relationships between input and output data in the given data stream. A concept in “concept drift” refers to the unknown and hidden relationship between inputs and output variables. The change to the data could take any form. Some other types of changes may include (i) a gradual change over time, (ii) a recurring or cyclical change, (iii) a sudden or abrupt change.

The learning models need to adapt to the changes quickly and accurately. The concept drift detection technique is applied for autonomous detection of incoming new traffic. The drift detector could be recognized as the simplest classifier, but it is not as simple as it looks like. The drift detection can replace the outdated models and reduce time, but on the other hand, it should not accept too many false alarms. Concept drift detector is an algorithm that detects the information about incoming signal and return signals about its changing patterns. Usually, after returning the signal about the drift, the model should be rebuilt as quickly as possible.

Author α σ ρ: Final year B.E, Department of CSE, Anna University, Chennai. e-mail: duripradheep@gmail.com
Author ω: Teaching Fellow, Department of CSE, Anna University Chennai.

The ability of a classifier to take on new information and evolving the classifier without retrained on the data set fully is known as incremental learning. Incremental learning has been successfully implemented for many problems, where the data is changing. The goal of incremental learning is learning new training samples to improve the classification quality. There are many incremental learning models used for changing data, but various challenges arise in those learning mechanisms. Support vector machines, ensemble method and clustering are commonly used for incremental learning. In this paper, we used ensemble incremental learning model such as hierarchical Bayesian parameter estimation of the drift diffusion model (HDDM) as the drift detector.

Transfer learning is a machine learning method where a model developed for a task is reused as the starting point for a model to be applied on a second or another task. A pre-trained source model is chosen from available models. It is a popular approach where pre-trained models are used to solve tasks of other models. The model must be better than a naive or existing models to ensure that some feature learning has been performed and the model fits on the source task can then be used as the starting point for another model.

II. Realted Work

Pavl et al. (2005) developed an experimental framework for the analysis and comparison of supervised (classification) and unsupervised learning (clustering) techniques for detecting malicious activities in the net-work. The supervised methods evaluated in their work include support vector machines, multilayer perceptron, k-nearest neighbor, and decision trees. The unsupervised algorithms include k-means clustering and single linkage clustering. They assumed that training and test data come from the same unknown distribution and they consider the case where the test data comes from new attack patterns. This scenario helps us understand how much an IDS can adapt its knowledge to new malicious patterns. This is often very essential for an IDS system. The results showed that the supervised algorithms show better classification accuracy with known attacks on the data. Among these algorithms, the decision tree algorithm had achieved the best results. However, if there are unseen attacks in the test data, then the detection rate of supervised methods decreases significantly. This is where the unsupervised techniques perform better as they do not show a significant difference in accuracy for seen and unseen attacks. The supervised techniques generally perform better compared to unsupervised methods.

Tavallaee et al. (2009) analyzed the entire KDD data set statistically and made a re-port of it. The analysis showed that there are two important issues in the data set which highly affect the performance of evaluated systems, and results in a very poor evaluation of anomaly detection approaches. The issues are redundant records and level of difficulty. To solve these issues, a new data set, NSL-KDD which consists of selected records of the complete KDD data set was proposed. The number of records in the train and test sets is reasonable, which makes it perfect to run the experiments on the complete set without the need to randomly select a small portion. Therefore, the evaluation results of different research works will be consistent. There were no duplicate records in the proposed test sets; therefore, the performance of the learners is not biased by the methods which have better detection rates on the frequent records. NSL-KDD is a data set that was suggested to solve some of the inherent problems of the KDD'99 data set. Although this new version of the KDD data set still suffers from some of the problems and may not be a perfect representative of existing real networks, because of the lack of public datasets for network-based IDSs. Furthermore, the number of records in the NSL-KDD train and test sets is reasonable. This advantage makes it affordable to run the experiments on the complete set without the need to randomly select a small portion. Evaluation results of different research work can be consistent and used to be a comparable one.

Zamani and Movahedi (2013) suggested that traditional intrusion detection and prevention techniques, like firewalls, access control mechanisms, and encryptions, have several limitations. Protecting networks and systems from increasingly complex attacks like denial of service are hard for existing IDS. Most of the systems built based on such techniques suffer from high false negative and false positive detection rates and the lack of continuously adapting to changing behaviors. In this paper, they divided the ML-based approaches to intrusion detection into two categories: approaches based on artificial intelligence (AI) techniques and approaches based on Computational Intelligence (CI) methods. Important CI methodologies are artificial neural networks, evolutionary computation, artificial immune systems, and fuzzy logic. The unsupervised algorithms include the k-means clustering and single linkage clustering. Both supervised and unsupervised learning is used in Artificial Intelligence techniques. They reviewed several influential algorithms for intrusion detection based on various machine learning techniques.

Biswas (2018) proposed a system in which a subset of features is selected using feature selection algorithms and then the set of selected features is used to train different types of classifiers. They proposed an IDS model that compares the performances of different combinations of classifiers and feature selection algorithm. The feature selection techniques are CFS, IGR, PCA, and minimum redundancy. Maximum-relevance and the classifiers are k-NN, DT, NN, SVM,

and NB used in this paper. It is difficult to choose one algorithm or the classifier over another to implement an intrusion detection system. He used a different mix of feature selection algorithms and classifiers because each of the classifiers and the feature selection algorithms have an advantage as well as disadvantages. The highest accuracy obtained in all the combinations is for IGR feature selection with k-NN. k-NN classifier produces better performance than others. The IGR feature selection method is better than the others.

Yuan et al. (2018) proposed a network intrusion detection approach combining concept drift detection and incremental learning. They performed various machine learning mechanisms for intrusion detection and proposed a new learning method called concept drift ensemble incremental learning. Three classifiers are used and then they are ensemble into one. They used a hierarchical Bayesian parameter estimation of the Drift Diffusion Model method based on Hoeffding inequality as a concept drift detector to detect an abnormality and then designed ensemble-based incremental learning for classification. They used KDD CUP 99 set data set to demonstrate the robustness of the intrusion detection approach. They compared the normal incremental learning with the concept drift ensemble incremental learning and recorded the findings. They proved that concept drift ensemble incremental learning have higher accuracy than traditional incremental learning.

Sun et al. (2018) suggested a new ensemble learning approach, namely, DTEL, for incremental learning with concept drift. Diversity and Transfer-based Ensemble Learning employs a diversity-based selection to preserve previously trained models. A pre-trained model was used for retraining the selected features instead of directly applied to all features. The preserved or pre-trained models were further adapted to the current concept through transfer learning. The main potential drawback of DTEL is more costly than the other methods. Despite this disadvantage could be minimized by parallel implementation of DTEL since it can be naturally parallelized, it is still worth investigating other methods to reduce the complexity of DTEL. Other base learners were investigated to compare with DTEL. They used this algorithm on several data set like Cover type, poker hand, electricity, CTR Prediction. The accuracy of the other algorithms was compared with DTEL and proved its superiority.

III. System Design

a) System Architecture

The system (Figure 1) takes the NSL-KDD data set for pre-processing. The pre-processing involves mapping, feature scaling, data clearing, encoding, data sampling, and feature selection. Finally, the selected feature is mapped to the data set and the data set is split for training and testing. The drift is detected using the HDDM and Hoeffding tree algorithm. Incremental ensemble learning uses three classifiers such as MLP classifier, Multinomial NB and SGD classifier for training. In transfer learning, the model is trained with features and saved. The pre-trained model is again used for training and performance is measured with the first model.

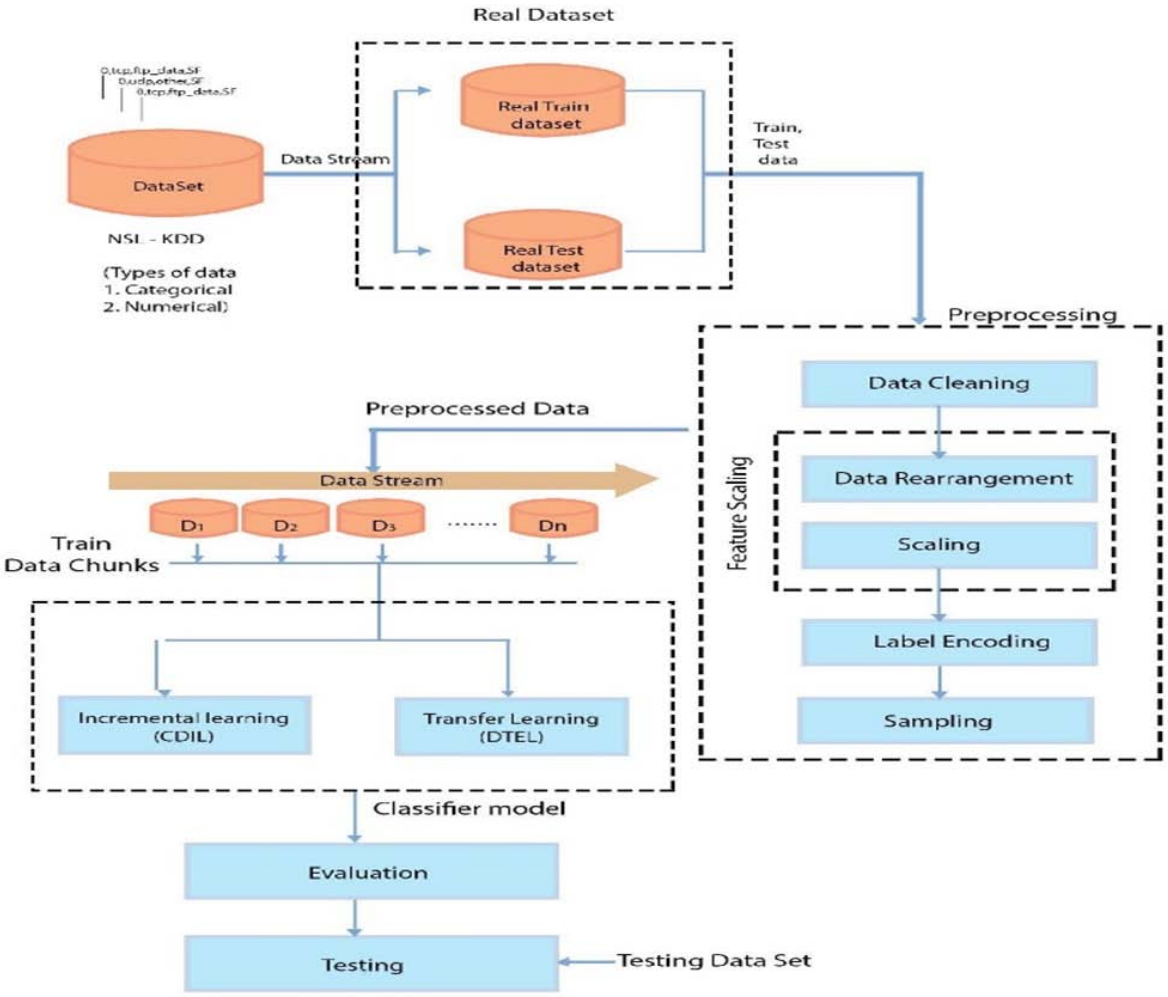

Figure 1: System Architecture

b) Data Processing

The data set is imported as a text document and then preprocessed. In preprocessing the data set is labeled and unwanted data columns are cleaned (Figure 2). The real data set which contains numerical data are scaled and characteristic data are encoded. The data set is sampled. It is followed by the feature selection process. In feature selection, only a few features are selected for training. After that, those features are mapped to the data set and other features are neglected. This proceeds with incremental and transfer learning.

c) Modules

i. *Data Processing*

First, the data set is labeled with the header and unwanted data columns with no values (i.e. fully occupied with zero) are dropped. The data set contains both numerical and characteristic data. The numerical data are scaled and character data are encoded. The data set is sampled for proper training. After sampling is done, it is followed by the feature selection process. In the feature selection process, only a few features that have a high influence on the target are selected for training.

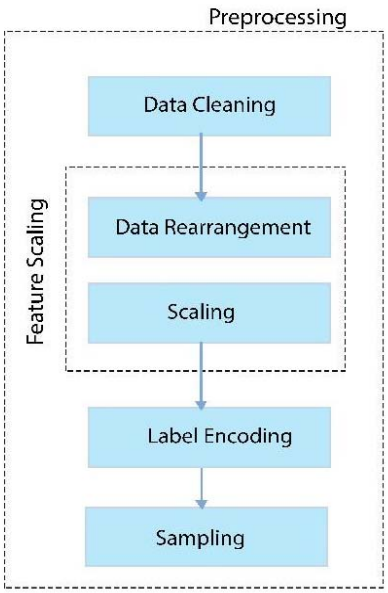

Figure 2: Data Processing

ii. *Incremental Learning (CDIL)*

Ensemble Incremental learning (Figure 3) helps to improve machine learning results by combining several models. Incremental learning keeps updating the model to generalize the model so that it doesn't deviate from the soul problem. Incremental learning is a machine learning mechanism where the learning process takes place whenever new examples or new attributes (attribute values) merge or deleted from the dataset and the solutions already obtained are only modified.

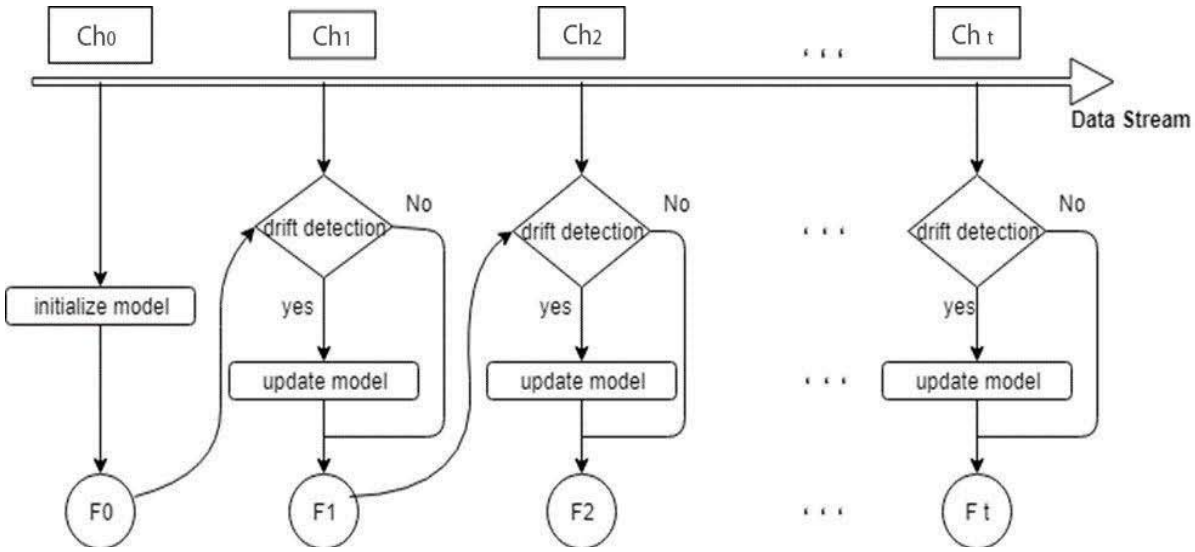

Figure 3: Incremental Learning

iii. *Transfer Learning*

The framework of transfer learning (Figure 4) differs from the other ensemble methods for incremental learning. First, it does not directly combine the outputs of historical models. Instead, each preserved historical model is first adapted to fit the current data, and then the adapted models and the model constructed from scratch are combined. This enables to achieve higher accuracy than the traditional method.

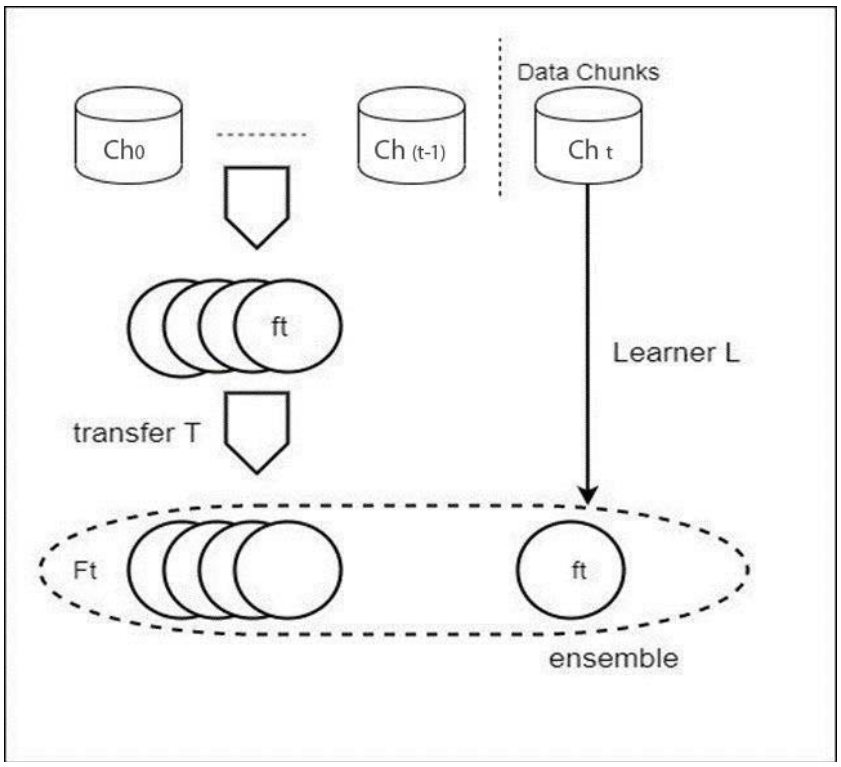

Figure 4: Transfer Learning

IV. System Development

a) Preprocessing

Large-scale data sets usually contain noisy, redundant and different types of data that present critical challenges to knowledge discovery and data modeling. Generally, the intrusion detection algorithms deal with one or more of the raw input data as numerical data only. Hence, we prepare data and convert categorical data in the dataset to numerical data. We found two important issues which highly affect the performance of evaluated systems, and results in a very poor evaluation of anomaly detection approaches. To solve these issues, we used the dataset NSL-KDD, which consists of the selected records of the complete KDD data set and does not suffer from any of mentioned shortcomings.

i. *Mapping*

The attacks in the data set were classified into 4 groups based on the data used by Tavallaee et al (2009). They are classified as Dos, R2L, Probe and U2R. But the dataset contains the attack as differentiated into 21 types of vulnerabilities. Hence, mapping in python is used to match all the attacks into five categories.

ii. *Scaling numerical attribute*

In machine learning, standardization is a key technique to get reliable results. Values for some features may diverge from small to very big numbers and the processes analyzed may explode the scale. Thus, all the values are scaled to a range.

iii. *Encoding*

We used the label encoding. In label encoding, we map each category to a number or a label. The labels chosen for the categories have no relationship. So categories are close to each other and lose such information after encoding.

iv. *Sampling*

Data sampling refers to statistical methods for selecting observations from the domain with the objective of estimating a population parameter. Data sampling is used to balance the data set. The recursive feature elimination is used to select the important feature with a high correlation.

v. *Feature selection*

Feature selection is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. We have 40 features in the data set hence we reduce them to nine features because it enables the machine learning algorithm to train faster. It reduces the complexity of a model and makes it easier to interpret.

b) Incremental learning

Algorithm 1:

Input: Preprocessed Data set

Output: Incremental model

1: from classifier.import hoeffding tree
2: from drift-detection.import hddm
3: Labels = Selected feature labels
4: Attributes = Selected feature attributes
5: Pairs = Hoeffding tree, hddm
6: Drift points = 20000, 40000, 60000, 80000
7: Acceptance interval = 250
8: Detection = Evaluate(Hoeffding tree, hddm, drift points, acceptance interval)

Algorithm 2:

Input: Data Chunk

Output: Drift Detected in data stream.

1: hddm (data set)
2: for i= length of data set do
3: Perform detection
4: if (detects drift point) then
5: Perform detection on next set
6: if (detects drift point) then
7: Update the model
8: else
9: Discard
10: end if
11: end if
12: end

c) Transfer learning

The model is trained with sequential classifier and then it is re-trained for n times.

Input: Preprocessed Data Set

Output: Classifier for data set

Algorithm:

1: from keras.models import Sequential
2: model= add (feature=n, hidden layer = k)
3: model= add (hidden layer = k-4)
4: Data set = model fit
5: Save the model
6: for i= fixed length do
7: for a = range of evaluation do
8: Load saved model
9: Retrain model (a)
10: end
11: end
12: evaluate

V. Results

This project used NSL-KDD data set which contains over 130000 data for training and 12000 data for testing.

In this paper, we first analyzed the dataset and plotted the distribution of the attack class (Figure 5). It gives a clear view of attack class distribution in both train and test data. The unwanted data are dropped. After this, we scaled the integer data into float and then the characteristic attributes into the integer. Because, machine learning can't be performed on the character data.

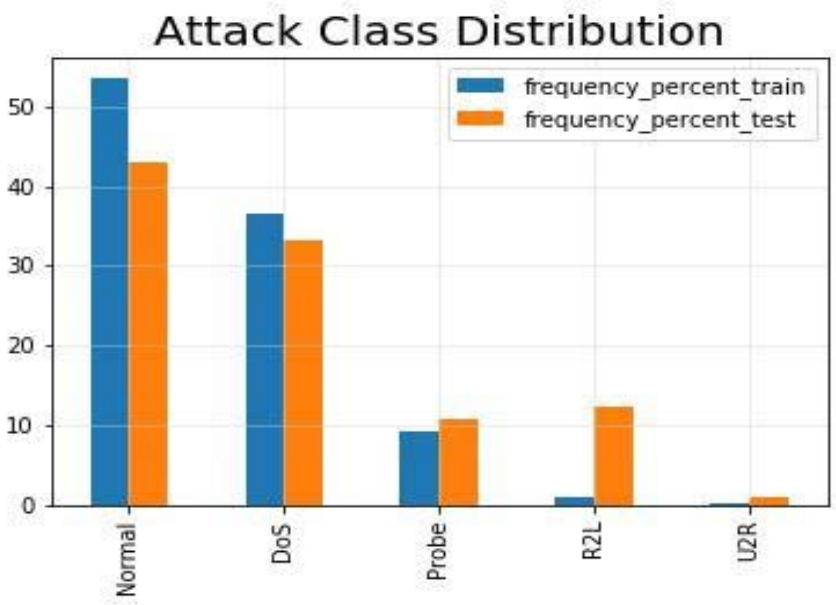

Figure 5: Attack Class Distribution

From this graph, we understand that, attack classes in test data is higher than the train data. This may affect learning. Hence, sampling is done and followed by feature selection. Feature selection is performed by the random forest classifier which sorts the features and it is plotted (Figure 6).

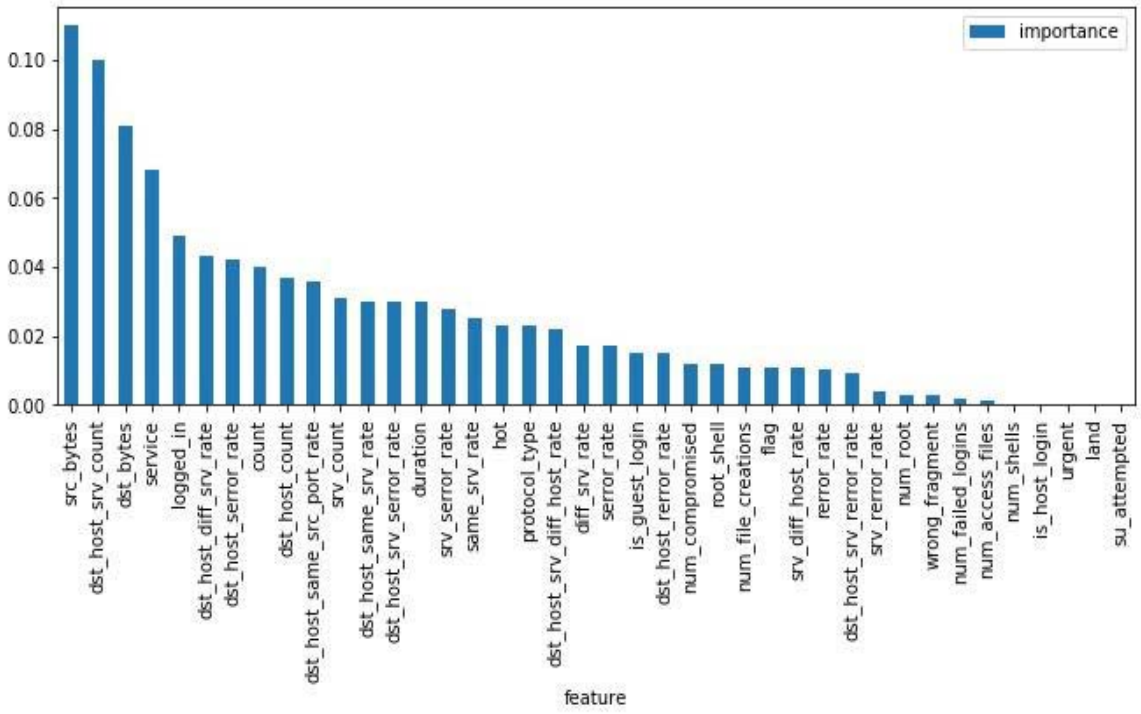

Figure 6: Features with high priority

From these first 10 features which have high priority is selected and those features are sorted in both train and test data set.

The drift is detected using the HDDM and Hoeffding tree algorithm. The peak points show the drift in the concept (Figure 7).

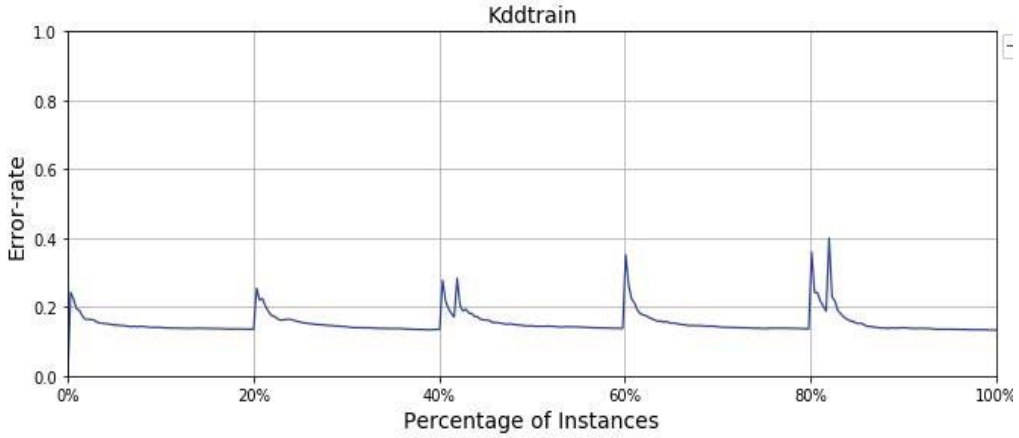

Figure 7: Drift Detection

Then in incremental ensemble learning classifiers such as MLP classifier, Multinomial NB and SGD classifier are used for training. The cross validation score is a statistical method used to estimate the performance of machine learning models. It is commonly used in machine learning to compare and select a model for a given predictive modeling problem because it is easy to understand and easy to implement. The cross-validation score is 0.880434910 which is +/- 0.003 of accuracy and shows that there is no overfitting.

Transfer learning makes use of the knowledge gained while solving one problem and applying it to a different but related problem. The knowledge of an already trained machine learning model is applied to related problem.

Table 1: Accuracy Analysis

	Incremental Learning	Transfer Learning
Accuracy	0.79%	0.88%

Higher accuracy is achieved with the transfer learning model (Table 1).

VI. Conclusion

In this paper, HDDM method is used based on Hoeffding inequality as a concept drift detector to detect an abnormality in the data chunks and then three classifiers are used to classify the intrusion. The transfer learning mechanism used in the proposed model shows higher accuracy than the incremental learning. The transfer learning used the knowledge from the previous learning and used it in the subsequent learning iteration. The evaluation results based on the NSL-KDD dataset demonstrate the intrusion detection approach. Hence the transfer learning can also be used in the intrusion detection system which contains sudden or abrupt concept drift.

References Références Referencias

1. Biswas, Saroj Kr. "Intrusion detection using machine learning: A comparison study." International Journal of Pure and Applied Mathematics 118, no. 19 (2018): 101-114.
2. Laskov, Pavel, Patrick Düssel, Christin Schäfer, and Konrad Rieck. "Learning intrusion detection: supervised or unsupervised?." In International Conference on Image Analysis and Processing, pp. 50-57. Springer, Berlin, Heidelberg, 2005.
3. Sun, Yu, Ke Tang, Zexuan Zhu, and Xin Yao. "Concept drift adaptation by exploiting historical knowledge." IEEE transactions on neural networks and learning systems 29, no. 10 (2018): 4822-4832.
4. Tavallaee, Mahbod, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani. "A detailed analysis of the KDD CUP 99 data set." In 2009 IEEE symposium on computational intelligence for security and defense applications, pp. 1-6. IEEE, 2009.
5. Yuan, Xiaoming, Ran Wang, Yi Zhuang, Kun Zhu, and Jie Hao. "A Concept Drift Based Ensemble Incremental Learning Approach for Intrusion Detection." In 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 350-357. IEEE, 2018.
6. Zamani, Mahdi, and Mahnush Movahedi. "Machine learning techniques for intrusion detection." arXiv preprint arXiv: 1312.2177 (2013).