ENHANCING DEEP LEARNING MODEL UNDER PARALLEL PROCESSING ON SPARK PLATFORM

Mr.A.Baskar[1]
Assistant Professor,CSE,
E.G.S.Pillay Engineering College,
Nagapattinam,India.

S.Roja[2]
P.G Scholar
E.G.S.Pillay Engineering College,
Nagapattinam,India.

Abstract*—Advancement in Internet produced a large amount of data. At the same time it leads to many attacks. These attacks are more complex and unpredictable. Intrusion Detection System which is shortly called as IDS is a necessary constituent for providing reliability in advanced Networks. Its design includes either designation-based detection or abnormality behavior detection. Lately, Researchers adopted Deep Learning (DL) mechanism. This mechanism produces a better Performance in respect to traditional Machine Learning Algorithms. Deep Learning in operation to build a Model for the IDS may take a prolonged time because of computation complexity and a large number of hyper parameters. Distinct Deep Learning models for IDS on Apache Spark have been imposed in this paper. This paper uses the popular Network Security Lab - Knowledge Discovery and Data Mining (NSL-KDD) dataset and shows a computation delay correlation between Apache Spark and regular implementation. Moreover, an extended model is used to enhance attack detection accuracy.*

***Keywords*—Intrusion detection, Apache Spark, Deep learning;**

I. INTRODUCTION

Computer networks have been boosted over the years, increasing to social and economic growth. The Internet Security Threat Report (ISTR) states that one in Thirteen Web requests is malicious software. The junk mails had increased to 55%, malware had risen to 46 %, and other network Threats. Cybercrime and threat actions have risen and have become a critical threat. IDS is an entry point software System to detect the Network attacks. This progress promoted an increase in network security. By inspecting packets obtained from the network, IDS helps to determine hazards. IDS is an entry point software to identify packets from the Network. It is also called as gateway. This overall improves Network Security. IDS System with traditional models already existed. In this project we are upgrading models by Deep learning Algorithms. This model mainly enhance the Internet Security.

1.1DEEP LEARNING

Deep learning is simply defined as the advanced version of machine learning. Machine learning is the one in which system learns by itself. System learns Independently without human support. Humans will train the system by feeding input and output data values. Finally by giving the input values system will predict the output values. There are many Traditional Machine Learning models to identify Network attacks, Since the Network attacks grows simultaneously we moved into the concept called Deep learning. Deep Learning is the up to date model of machine learning to detect different type of attacks. Detection accuracy produced by Deep Learning model is high compared to Machine Learning Models. This learning predicts Multi class classification. Deep Learning (DL) is used for various applications in many areas such as Graphics Processing, Artificial Intelligence, Computer Perception…

1.2APACHE SPARK

IDS uses Apache Spark for processing large volume of data. Apache Spark is easy for accessing. Spark is an engine which works fast for large Big data processing. Simultaneously Spark executes the operations which is stored in memory.

Apache Spark is a free source software, disperse processing system which is handed-down for Big Data workloads. It follows Bottom-Up approach for performing data operations and process the data parallely within short interval of time. It is superior than Hadoop for big data scale processing. It works 100 times greater than Hadoop. We use Apache Spark platform to reduce time complexity. Apache Spark is used for Parallel Classifiers for processing large volume of data with reduced interval of time.

1.3NSL-KDD DATASET

NSL KDD Dataset is used for Training input and Output Variables. NSL-KDD is a standard data set recommended to solve some of the inborn problems of the KDD99 data set. This dataset is the popular dataset which doesn't contain Redundant Values. It is the Redefined Version of KDD99 Dataset.KDD99 Dataset doesn't contain duplicate values. NSL KDD Dataset predict five attacks. While UMSW NV15 Dataset predict fourteen attacks. In this project we Use NSL KDD Dataset. The records found in the training and test sets are Calculable.

II. PREVIOUS WORK

Many Papers Published Machine Learning and there are Traditional Machine Learning Models to Predict the Network attacks. Already existed models predict the attacks with less accuracy. There are class imbalance problems as a result it predicts the Network attacks with

48% accuracy. Since there are different Mode of Network Connection namely, Dial-up Networks, Virtual Private Networks, Local Area Networks, Direct networks, Mobile Networks and Fiber Optic Cables… Network attacks grows increasingly. Moreover Time Complexity for processing data is also increases. So the Existing Model has to be upgraded. In Machine Learning Different Algorithms are used to improve accuracy but the accuracy is less due to class imbalance. **Neural Network and LDA algorithm** produces high accuracy when compared with other algorithms. Overall it produces accuracy with (**46%-48%)** compared with other algorithms.

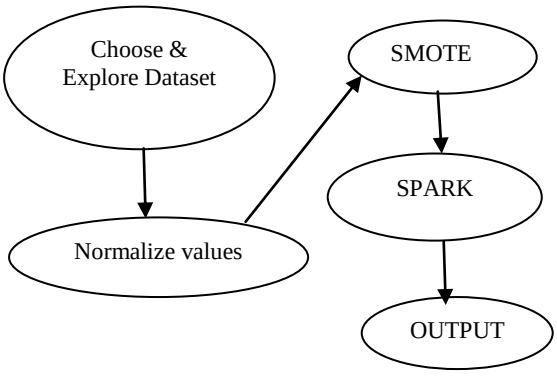
III. PRESENT WORK

In Proposed Model, Deep learning Model is used in order to improve accuracy than previous models. Working on Deep Learning Algorithms to enhance model development. In the first approach we implement all Deep Learning algorithms. **Adagrad method** algorithm produces high accuracy with **47%** over other algorithms. On analysing it is less accuracy compared with Machine Learning models. In the second method, to improve accuracy we use hybrid model which is a combination of LSTM model with advanced technique of SMOTE. Using SMOTE Sampling technique Accuracy has been increased with **75.7%** accuracy. But it takes more time for Processing. In the third method Parallel Classifiers are used to improve time complexity. Apache Spark Model is used for parallel processing. On Processing the time complexity is reduced and the overall time taken for processing is **0:00:02.2344.**The objectives is as follows Propose a Deep learning-based Intrusion detection system with more accuracy compared with already developed system models. Use Spark Cluster configuration to minimize the training process at the mean time implementing the IDS with different hyper parameters. Solving challenges related to the selected dataset, NSL-KDD, such as class imbalance. A hybrid solution has been introduced to solve class imbalance.

IV. SYSTEM MODEL

In this Model Preprocessing is done in the earlier stage to convert all values into numeric values. Preprocessing is done to avoid duplicate values, Null values are also eliminated in Preprocessing. Normalize Values to find Minimum and Maximum Values. During Training set X and Y values are given to the system and trained accordingly. In testing phase X values are given and the system has to predict Y values on learning by itself. SMOTE, a Sampling technique is used in the combination of LSTM model to predict Y values with high accuracy. For Class Imbalance SMOTE Technique is used as a solution **Refer Fig (1.1)**. Spark Model is used after preprocessing and SMOTE Technique for parallel processing. The Processing Speed increases with reduced time. On Reference **[6]** IDS Model Block has been established. This reference gives the basic idea model of Intrusion Detection System. In the first step dataset is chosen and explored. Dataset is nothing but a collection of data's. Dataset is then trained and this phase is called as Training Dataset. In the second step Preprocessing is done and converting Packet nominal values to numeric values. This Process eliminates null values. Sorting values from Minimum to maximum values. Entering into the third step **SMOTE** Technique is used as a solution for class imbalance and to improve accuracy. In Final phase **SPARK** Model has been introduced to produce output within a short interval of time. Hence Time complexity has been reduced in this stage.

4.1 IDS Model Block Diagram



4.2 SPARK Architecture

Spark Architecture consists of mainly three parts such as spark driver, cluster and Manager In **fig4.2** First Clustering is done which divides the similar data into the same groups then, Manager Splits the data into multiple workers and workload is shared by multiple workers. Then the task is handled by Parallel Processors and Parallel Processing is done in this stage. It is the advanced version of sequential processing. Finally Output is produced by parallel workers and combined

into a single output with high accuracy and reduced time complexity. Higher level Application Programming interface is used. In this Paper Multi-Class Classification is used to detect Network attacks namely as follows, Normal attack, Denial of service attack, R2L, Probe and U2R.

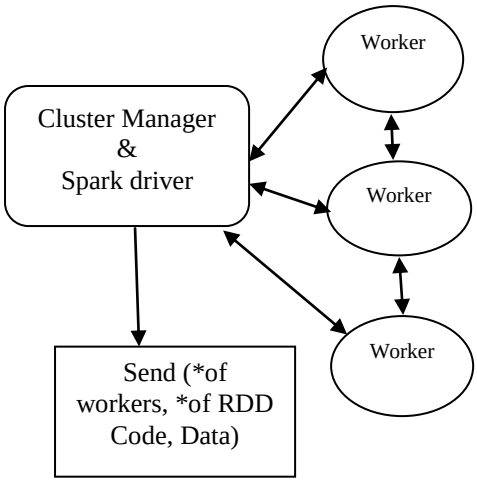


Fig.4.2 Spark Architecture Diagram

4.3 PERFORMANCE MEASURE

4.3.1 Accuracy
It is the ratio of the correctly classified packets to the total dataset.

$$\frac{TP+TN}{TP+TN+FP+FN}$$

4.3.2 Precision

It is the ratio of correctly classified attacks to the total number of identified attacks. It can be calculated as:

$$\frac{TP}{TP+FP}$$

4.3.3 Recall

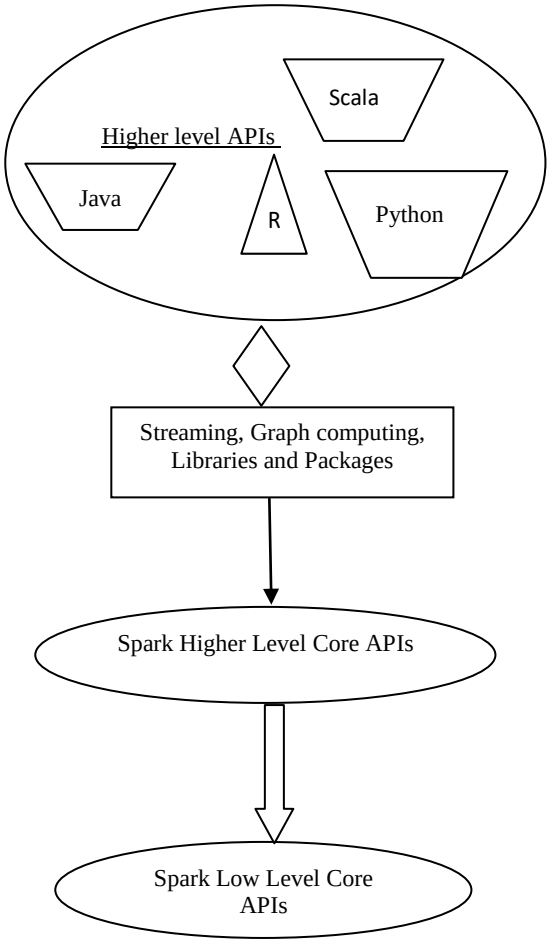
It is the ratio of accurately classified attacks to the total number of attacks in the test dataset. It can be calculated as:

$$\frac{TP}{TP+FN}$$

V. SYSTEM INTERFACE

An interface is an interconnection between two machines. It is an interconnection between a system and humans or between two hardware's. It is simply termed as exchange of information. The exchange can be either between Computer hardware, System Software or peripheral devices. An interface Diagram has been illustrated below on Fig.4.3

Fig.5.1 Interface Diagram



5.2 COMPARISON

On Training and Test Set Comparison Type of attack category are given in X axis and Values i.e. No of records are given in Y axis. The number of records and values varies accordingly for each type of attack categories.

For instance R2L Remote to Local attack on training set Produces Zero value while on the test set produces a range of 3000 records. Similarly Normal attacks also varies in training and test set. Normal attacks gives highest range of values in training set while in test set it gives middle range of values. Probe attack gives low range of values in training and on the other end produces middle range of values.Deniel of Service attack (DOS) varies accordingly in training and test phase. User to Root attack (U2R) also gives a major different set of values while plotting a graph.

VI. IMPLEMENTATION & OUTPUT

MACHINE LEARNING ACCURACY

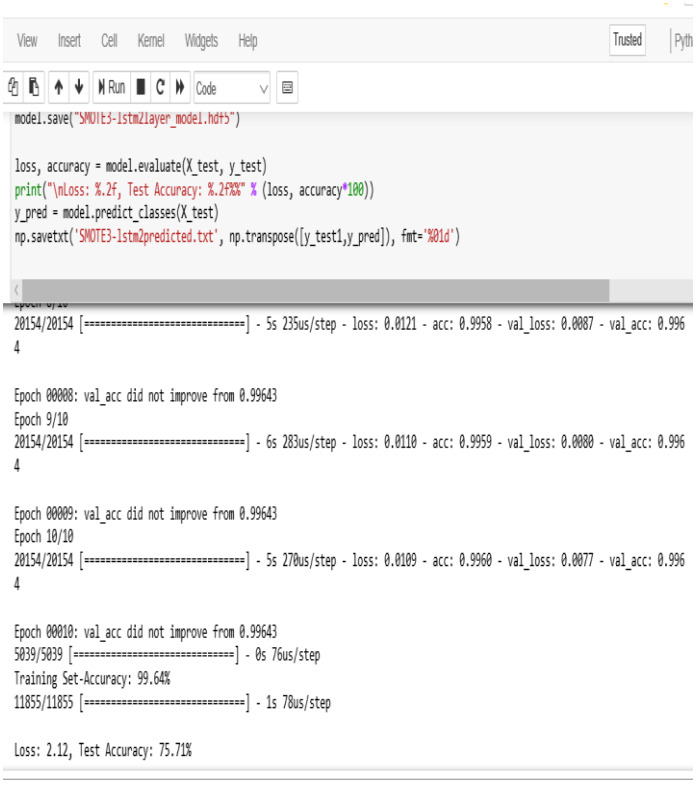
```
algorithm=['NB','DT','LR','KNN','Neural NW','SVM','SGD','LDA']
Training=[round(t1,2),round(t2,2),round(t3,2),round(t4,2),round(t5,2),round(t6,2
Test=[round(avg),round(avg1),round(avg2),round(avg3),round(avg4),round(avg5),rou

df = pd.DataFrame({'Test': Test,'Training Set': Training}, index=algorithm)
df.T
```

```
C:\Users\rose\Anaconda3\envs\roja_tfd\lib\site-packages\ipykernel_launcher.py:14
ved in a future version. Use .values instead.
```

```
NB-Test Set-Accuracy is  29.44749051033319
DT-Test Set-Accuracy is  44.88401518346689
LR-Test Set-Accuracy is  38.71784057359764
KNN-Test Set-Accuracy is  43.90552509489667
Neural NW-Test Set-Accuracy is  46.90004217629692
SVM-Test Set-Accuracy is  39.730071699704766
SGD-Test Set-Accuracy is  23.736819907212148
LDA-Test Set-Accuracy is  47.20371151412906
```

IMBALANCE DATASET ALL ALGORITHMS

DEEP LEARNING ACCURACY



```
model.save("SMOTE3-lstm2layer_model.hdf5")

loss, accuracy = model.evaluate(X_test, y_test)
print("\nLoss: %.2f, Test Accuracy: %.2f%%" % (loss, accuracy*100))
y_pred = model.predict_classes(X_test)
np.savetxt('SMOTE3-lstm2predicted.txt', np.transpose([y_test1,y_pred]), fmt='%01d')
```

```
20154/20154 [==============================] - 5s 235us/step - loss: 0.0121 - acc: 0.9958 - val_loss: 0.0087 - val_acc: 0.996
4

Epoch 00008: val_acc did not improve from 0.99643
Epoch 9/10
20154/20154 [==============================] - 6s 283us/step - loss: 0.0110 - acc: 0.9959 - val_loss: 0.0080 - val_acc: 0.996
4

Epoch 00009: val_acc did not improve from 0.99643
Epoch 10/10
20154/20154 [==============================] - 5s 270us/step - loss: 0.0109 - acc: 0.9960 - val_loss: 0.0077 - val_acc: 0.996
4

Epoch 00010: val_acc did not improve from 0.99643
5039/5039 [==============================] - 0s 76us/step
Training Set-Accuracy: 99.64%
11855/11855 [==============================] - 1s 78us/step

Loss: 2.12, Test Accuracy: 75.71%
```

SPARK MODEL

```
pipe = Pipeline(stages=[assembler, label_indexer])
pipe_model = pipe.fit(iris1)
data = pipe_model.transform(iris1)
data = data.select("features","label")
```

```
In [31]: data
```

```
Out[31]: DataFrame[features: vector, label: double]
```

```
In [33]: import math
         import datetime
         import time
         start=datetime.datetime.now()
         predictiondt = modeldt.transform(data)
         predictiondt.toPandas().head()
         end=datetime.datetime.now()
         elapsed=end-start
         print('time taken-spark Model:',elapsed)
```

```
time taken-spark Model: 0:00:02.234483
```

VII. CONCLUSION

This paper proposed a Deep Learning Model for advanced Network attacks with high accuracy over traditional models. At the Same time Parallel Classifiers method is used to predict output within a short interval of time. Comparing Different Outputs by various algorithms. First in Machine Learning By implementing all algorithms we found Neural Network and LDA Produces high accuracy of **48%** over others. In the second phase, Moving into Deep Learning Model LSTM and SMOTE algorithms predicts the output values with **77%** accuracy over Machine Learning. Finally Apache Spark Model is used to reduce time complexity by parallel classification and predicts the result output within a few seconds.

REFERENCES

[1] Symantec Corporation, "2018 Internet security threat report," vol. 23, pp. 1–89, 2018.

[2] A. A. Ghorbani, W. Lu, and M. Tavallaee, "Network Intrusion Detection and Prevention," vol. 47, pp. 27–54, 2010.

[3] E. Hodo, X. Bellekens, A. Hamilton, C. Tachtatzis, and R. Atkinson, "Shallow and Deep Networks Intrusion Detection System: A Taxonomy and Survey," pp. 1–43, 2017.

[4] M. A. Alsheikh, D. Niyato, S. Lin, H. P. Tan, and Z. Han, "Mobile big data analytics using deep learning and apache spark," IEEE Netw., vol. 30, no. 3, pp. 22–29, 2016.

[5] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, "A survey of deep learning-based network anomaly detection," Cluster Comput., pp. 1–13, 2017.

[6]A. Chu, Y. Lai, and J. Liu, "Industrial Control Intrusion Detection Approach Based on Multiclassification GoogLeNet-LSTM Model," Secur. Commun. Networks, vol. 2019, no. 2, 2019. [7] S. H. Khan, M. Hayat, and F. Porikli, "Regularization of deep neural networks with spectral dropout," Neural Networks, vol. 110, pp. 82–90, 2019.

[8] H. Alaiz-Moreton, J. Aveleira-Mata, J.OndicolGarcia, A. L. Muñoz-Castañeda, I. García, and C. Benavides, "Multiclass Classification Procedure for Detecting Attacks on MQTT-IoT Protocol," Complexity, vol. 2019, 2019.

[9] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1– 15, 2015.

[10] J. C. Bansal, N. Delhi, K. Deep, and A. K. Nagar, Evolutionary Machine Learning Techniques, no. January. Springer Singapore, 2020.