Intrusion Detection System on KDD'99 Dataset with Imbalanced Classes

Anupam Agrawal

***Abstract**: The paper describes a method of intrusion detection that keeps check of it with help of machine learning algorithms. The experiments have been conducted over KDD'99 cup dataset, which is an imbalanced dataset, cause of which recall of some classes coming drastically low as there were not enough instances of it in there. For Preprocessing of dataset One Hot Encoding and Label Encoding to make it machine readable. The dimensionality of dataset has been reduced using Principal Component Analysis and classification of dataset into classes viz. attack and normal is done by Naïve Bayes Classifier. Due to imbalanced nature, shift of focus was on recall and overall recall and compared with other models which have achieved great accuracy. Based on the results, using a self optimizing loop, model has achieved better geometric mean accuracy.*

***Keywords**: Intrusion detection system, One Hot Encoding, Imbalanced classification, Geometric Mean Accuracy.*

I. INTRODUCTION

This paper presents the check to the arena where the technology being used for connecting people and more and more features introducing on the side. The more complex structure it is in the backend and that can possibly lead to possible chances of threat from masqueraders and phishers, whom with data would not be secured. [1]There are basic network security which enables the network to communicate with a set password and some more advanced techniques like firewall encryption but this does not cater the needs for full secure network system as they miss out on some of the important basics and attacks which can be proven hazardous to system. Intrusion Detection this method is very viable in scrutinizing and guarding the network or information. It is able to detect any unsuitable, suspicious activity. An intrusion detection system is the most essential part for devices which are being used for networking and communication as to ensure security and stability. IDS has been used mainly based on network to detect, identify and track the intruders attacking the network or system.

II. LITERATURE SURVEY

Beginning in 1980, [1]Intrusion detection was born. Since then, the system has undergone several advancements and evolved till its present state. Intrusion Detection System (IDS) acts as a defensive tool and nowadays web application security is one of the important issue to focus on due to growth in number of web attacks. According to the Internet Security Threat Report(ISTR) 2017, more than 76% of sites taken into consideration were vulnerable. A survey reports that most of the time hacker attacks either targeted the web applications or utilized them as the attack vectors. W. Wang et al.(2020) [2] proposed a novel intrusion detection system based on an optimal hybrid kernel extreme machine learning machine model for KDD dataset. They also tabulated the class wise precision of different methods(models) proposed by researchers till now.

Table-I: Geometric mean of recalls with various methods

Method	Classes					
	Normal	Dos	Probe	U2R	R2L	Geometric Mean
KDDwinner	99.45	97.12	83.32	13.16	8.40	38.89
SVM	99.30	99.50	97.50	19.70	28.80	55.91
CSVAC	99.91	99.72	65.74	42.59	20.47	56.41
CPSO-SVM	96.87	99.98	63.61	11.08	50.27	50.94
RTMAS-AIDS	97.89	99.79	91.86	24.68	35.90	60.27

We can observe from this table that the geometric mean of recalls is not satisfactory. The precision of inferior classes namely U2R and R2L are very low.

W. L. Al-Yaseen et al. (2017)[3] proposed a Multi-level hybrid support vector machine and extreme learning machine model based on modified K-means for detection systems. The recalls of the respective classes achieved by their model were having the geometric mean of recall 52.64%. We explored various papers, there was significantly less geometric mean of recalls or not satisfactory recall values for rare attacks. We explored a Decision Tree model tuned with customized parameters which has up to mark recall values for each attack and hence a better geometric mean.

III. SYSTEM OVERVIEW

A. Dataset

This database contains a standard set of data[4] to be valued and calculated, which includes a huge examples of attacks simulated in a secure system of network.[5].

Although the KDD99 dataset is more than 15 years old, the number of research works uses it is the most in area of machine learning and IDS, and it is the de facto dataset for these research areas.

Manuscript received on October 26, 2021.
Revised Manuscript received on November 01, 2021.
Manuscript published on December 30, 2021.
* Correspondence Author

Anupam Agrawal*, Department of Computer Science, Maulana Azad National Institute of Technology (MANIT), Bhopal (M.P) India.

B. Preprocessing

- *Label Encoding* Label Encoding of columns with string values into number form which can be read by machine.
- *One Hot Encoding* One hot encoding is a process in which categorical variables are converted into a form that can be fed to Machine learning algorithms so that it can do a better job in prediction. For example, one column has three features namely {0,1,2} is replaced by three columns putting only that value 1 where the feature is present and then further remove the first column so that to get more optimized dataset.

C. Feature Reduction

- *PCA:* Principal Component Analysis reduces dimension but when dealing with multi-class data it's necessary to reduce dimensions in a way that inter class separation is also taken care of.

Advantages of PCA

1. It search for liner combination of variables that efficiently separates two classes.
2. It reduces the measure of overfitting.

D. Classification and Regression Trees

Decision tree learning is one of the methods used in statistics, data mining and machine learning for prediction. It uses a decision tree (as a predictive model) to go from observations about an item that is represented in the branches to conclusions about the item's target value that is represented in leaves.

Tree models where the target values can take a different values are called classification trees; in these structures, leaves represent class labels and branches signifies the put together of those features that will result to class labels.

- *Parameter of Classification and Regression trees*

1. Stopping Criterion - This classification is recursive binary split and the model should be trained to stop split while it goes down with training data. The most used stopping criteria is minimum count on the number of training instances assigned to each leaf node. The model will stop right there where it get the minimum split and that node is accepted as final leaf node.

It defines to the training data how the model will be. Too specific and the tree will overfit the training data and likely have poor performance.

2. Class Weights – The tree is constructed for variables in the dataset by splitting the training dataset using values. It provides the weightage to classes to balance out the imbalance of the dataset. At each node, the each breaking down in the data results in the more refined group of examples is chosen in a greedy manner.

The splitting criterion can be updated to not only take the accuracy of the split into account, but also to be weighted by the importance of each class.

- *Small Weight*: Less Importance, lower influence on node correctness.
- *Large Weight*: It has more importance and influence more on node correctness.

The scikit-learn Python library supports implementing decision tree algorithm that supports class weighting.

It has a class-weight argument that acts as model hypermeter. The class-weight terms each class label and the weighting to apply in the calculation of group purity for splits in the splits in the decision tree when fitting the model.

The class weighing has been defined by many ways: say

- Tuning determined by a hypermeter search such as a grid search
- Heuristic specified using a general best practice

Inverse of class distribution has been used here as a class weighting practice.

E. Metrices:

To evaluate metrics[6] there has been many ways,

1. Confusion Matrix – One of the key concepts in classification performance, it helps to visualize the model predictions versus the ground truth labels. Each row of confusion matrix presents the values in a predicted class and each column represents the examples in actual class.

- *Classification Accuracy*

  Classification accuracy is defined as number of correct predictions divided by the total number of predictions multiplied by 100. It tells us about the how accurate system works.

- *Precision*

  When the class distribution is imbalanced, that leads to prediction of all samples to the particular occurring class, we get a high accuracy rate, which becomes partial and thus cannot rely on that. If we look at class wise precision when calculated by formula(1) and model has high precision for classes with high frequency.

- *Recall*

  It is defined as fraction which model predicted the class accurately out of all the instances it predict the class. Hence in this model recall defined the model accuracy more but still with imbalanced data it cannot suffice alone and it needs some additional weightage.

- *F1 Score*

  Depends on application, we may want to give higher priority to recall or precision. But as on some conditions precision can be more accurately represent the model or recall. Thus in f1 score it combines both into a single metric. F1-score is defined as the Harmonic Mean(HM) of recall and precision.

In case of multi_class classification, one will get metrics for each class. To check the overall quality of classifier, we take mean of precision of all classes or mean of recall of all classes.

Mean precision of model = Mean(method(c) (1)

c: classes in model

method: precision or recall

This mean can be weighted as simple arithmetic, geometric or harmonic. Geometric mean is more deviated towards least value.

For example: Arithmetic mean(2,32) = 17 while Geometric mean(2,32) = 8.

Generally in a balanced dataset classifier, preference given to the arithmetic mean as it is less complex to calculate. But for an imbalanced classification, we must take the geometric mean. As this data is more of imbalanced classification and hence results will be overall geometric mean of recall of all classes.

IV. EXPERIMENTS AND THEIR RESULTS

For different Decision Tree Classifier (DTC), as data fed to the different models and:

A. Simple Decision Tree Classifier

Formula applied for the classifier for calculating the values for classes
Classifier = DTC (criterion='entropy',random_state=0) (1)

Table II: Geometric mean of Simple DTC of classes

0.9947683	0.9725824	0.7148343	0.0368768	0.1491228

Overall GM of recall : 0.328118292662357

B. Decision Tree Classifier with balanced weights :

Formula applied for the classifier for calculating the values for classes.
Classifier = DTC (criterion='entropy', random_state=0, class_weight='balanced') (2)

Table III : Geometric mean of DTC with balanced weights

0.98395854	0.97292182	0.7774844	0.0285379	0.0877193

Overall GM of recall : 0.28448063318630257

C. Decision Tree Classifier with balanced weights and custom min_samples_leaf :

Formula applied for the classifier for calculating the values for classes
Classifier = DTC (criterion='entropy', random_state=0 min_samples_leaf=1000, class_weight='balanced') (3)

Table IV : Geometric mean of DTC with balanced weights and min_samples_leaf

0.9649464	0.8092650	0.7762842	0.2704923	0.2368421

Overall GM of recall : 0.5222106400458261

It can be observed that using balanced weights and min_samples leaf criterion, the geometric mean of recall is improved considerably.

- **Self optimizing model :**
  KDD dataset is too unbalanced. There are new types of attacks which are absent in the training dataset. To improve the predictions further, the adoption of self optimizing model happened in which class_weights are updated on the basis of current results. The class_weight of class with minimum recall is multiplied by constant K=1.5 in each iteration.
- **Random weights model:**
  After trying multiple random weights models, the following model has best results;

Classifier = DTC(criterion='entropy', random_state=0, min_samples_leaf=4000, class_weight={0:0.1, 1:1, 2:100, 3:10000, 4:8000}) (4)

Table IV: Training dataset used in model

5267	171	13522	16045	14973
0	384117	722	6162	457
0	4	3785	302	16
0	0	0	1116	10
0	0	0	11	41

Accuracy = 0.8939417555124175
Recall = 0.8248688104925828
Precision = 0.12125637294478411

Table V: Testing dataset results

43064	35	9149	4413	3932
6	218970	1433	9162	282
1	68	3058	1012	27
8155	5	1985	6004	40
0	1	23	34	170

Accuracy = 0.8721566156210514
Recall = 0.6723816626775823
Precision = 0.2833249999945715

V. CONCLUSION AND FUTURE SCOPE

In this research, by tuning the class weights, we have built the CART model for KDD dataset and achieved the geometric mean of recalls close to 67.2% which is a significantly improved result with a fine approach. The R2L has minimal detection rate when comparing with that of the other categories because some of the attacks included in R2L, such as "snmpgeattack" (7741/16189) and "snmpguess" (2406/16189), exhibit features that are highly similar to those of Normal and may match these features 100% such that they cannot be 100% classified as attacks. The low detection rate for U2R can be happening due to the very small number of examples or data (288) when comparing with that of other. Some of the attacks in U2R has characteristics that has same in some examples of the Normal category[8]. In some models, other machine learning models like Area Under Curve(AUC). Extensive Machine Learning (ELM) on the basis of class weights. Using hybrid systems as proposed by researchers can be beneficial for training hierarchical model and will be used for detecting much rare attacks.

REFERENCES

1. J. P. Anderson, "Computer Security Threat Monitoring and Surveillance," Technical Report, James P. Anderson Company, Fort Wahington, 1980.
2. L. Lv, W. Wang, Z. Zhang, and X. Liu, "A novel intrusion detection system based on an optimal hybrid kernel extreme learning machine," Knowledge-Based Systems, vol. 195, p.105648, 2020.
3. W. L. Al-Yassen, 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. 296-302,2017.
4. KDD Cup 1999 Data. [Online] Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
5. H.G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, "A hierarchical SOM-based intrusion detection system," Engineering Applications of Artificial Intelligence, vol. 20, no. 4, pp. 439-451, 2007.
6. Boyle, Tara. "Methods for Dealing with Imbalanced Data." Medium, Towards Data Science, 4 Feb 2019. Available: towardsdatascience.com/methods-fordealing-with-imbalanced-data-5b761be45a18.
7. R. Beghdad, "Efficient deterministic method for detecting new U2R attacks," Computer Communications, vol. 32, no. 6, pp. 1104-1110,2009
8. Multi-Level Hybrid Support Vector Machine and Extreme Learning Machine Based on Modified K-means for Intrusion Detection System Wathiq Laftah Al-Yaseen , Zulaiha Ali Othman , Mohd Zakree Ahmad Nazri.

AUTHORS PROFILE

Anupam Agrawal, has graduated from NIT Bhopal with Computer Science as majors. An enthusiast of Fin-tech and social deeds. He has worked on projects like to alleviate the socially challenged people. He is an active member of Aaroha, social club and volunteer for under privileged children education and completely believe as education is the one truly best investment on human capital. He takes interest in playing badminton and make part time short movies and videos on Youtube. He plans to shape his career in a way where he can give back to the society and serve for the humanity and compassion.