Intrusion Detection Using Machine Learning

Susmitha Kolli
Electrical Engineering
San Jose State University
San Jose, CA
susmitha.kolli@sjsu.edu

Mahantesh Shashikant Mise
Electrical Engineering
San Jose State University
San Jose, CA
mahantesh.mise@sjsu.edu

Mamatha Ramesh
Computer Engineering
San Jose State University
San Jose, CA
mamatha.ramesh@sjsu.edu

ABSTRACT

The primary intention of this project is to perform intrusion detection and network security analysis using machine learning. Intrusion detection defines the detection of unwanted traffic on a network. The unwanted traffic may be in the form of malicious activities or policy violations. The project works on the standard KDD Dataset. The obtained dataset is not in proper format and needs a lot of data cleaning and feature engineering. The designed algorithm learns from the training dataset which is initially provided. The classification is done using four classifiers, and the best performing classifier is pickled to make an intrusion detection application. The classifier segregates the data into one of the four intrusion types. The success of the project gets gauged from the confusion matrix and accuracy metrics such as accuracy, precision, recall and F1 score. The project demonstrates the effective use of the machine learning algorithm for intrusion detection in networks security.

KEYWORDS

Intrusion detection, Network Security, Machine Learning, Malicious activity, KDD Dataset, Data Cleansing, Feature Engineering, Classifiers, Confusion matrix, Accuracy.

1 Introduction

1.1 Network Security

Network Security deals with protecting the networks, computers, and data from attack, unauthorized access, change, and destruction [2]. Computer Security and Network Security are the building blocks of the cybersecurity system. The security system mainly consists of a firewall, anti-virus software, and Intrusion detection system [11]. IDS help to detect internal and external intrusions in the form of discover, analyze, identify unauthorized use, duplication, alteration and destruction of information systems [2].

IDS is classified into three types.

1. Misuse-based: Used to detect attacks based on the signature of the attack patterns. Misuse-based techniques don't help in detecting Zero-day attacks.
2. Anomaly-based: Works by identifying the anomalies as the deviations from the normal behavior. They have the capacity to detect the zero-day attacks.
3. Hybrid techniques: They use the combination of the Misuse-based and Anomaly-based detection techniques. The hybrid technique helps to decrease the false positives for unknown attacks.

1.2 Machine learning

Training and testing are the two critical phases of machine learning. The process of learning mainly consists of four steps.

1. Identifying classes and attributes from the training data.
2. Identifying the attributes for classification through modification, redundancy check, and feature engineering.
3. Decide on the classifiers to be applied to the reduced data set.
4. Classify real-time unknown dataset using a trained model.

To increase the accuracy of the obtained result, the project introduces another important step of validation to determine the best classifier algorithm to be used to compare the output result. The comparison helps the algorithm to predict the results more efficiently.

1.3 Dataset

The standard data set utilized in the project is KDD CUP 99. The KDD CUP 99 consists of nine weeks of raw TCP dump data of LAN (Local Area Network) of a U.S. Air Force peppered with multiple attacks [8]. The size of the raw TCP dump data collected over 4 weeks was of size 4 bytes. Five million connection records were created using the above data capture. Each network connection consists of well-defined [3] data flows from source IP address to a destination IP [12] address using necessary protocols. Table1[4] presents various features present in KDD Cup 99 dataset.
Table 1: List of KDD Cup '99 features and their descriptions

No	Feature	Description
1	duration	Duration of the connection
2	protocol type	Connection protocol (e.g. TCP, UDP, ICMP)
3	service	Destination service
4	flag	Status flag of the connection
5	source bytes	Bytes sent from source to destination
6	destination bytes	Bytes sent from destination to source
7	land	1 if connection is from/to the same host/port; 0 otherwise
8	wrong fragment	Number of wrong fragments
9	urgent	Number of urgent packets
10	hot	Number of "hot" indicators
11	failed logins	Number of failed logins
12	logged in	1 if successfully logged in; 0 otherwise
13	#compromised	Number of "compromised" conditions
14	root shell	1 if root shell is obtained; 0 otherwise
15	su attempted	1 if "su root" command attempted; 0 otherwise
16	#root	Number of "root" accesses
17	#file creations	Number of file creation operations
18	#shells	Number of shell prompts
19	#access files	Number of operations on access control files
20	#outbound cmds	Number of outbound commands in a ftp session
21	is hot login	1 if login belongs to the "hot" list; 0 otherwise
22	is guest login	1 if the login is the "guest" login; 0 otherwise
23	count	Number of connections to the same host as the current connection in the past 2 seconds
24	srv count	Number of connections to the same service as the current connection in the past two seconds
25	serror rate	% of connections that have "SYN" errors
26	srv serror rate	% of connections that have "SYN" errors
27	rerror rate	% of connections that have REJ errors
28	srv rerror rate	% of connections that have REJ errors
29	same srv rate	% of connections to the same service
30	diff srv rate	% of connections to different services
31	srv diff host rate	% of connections to different hosts
32	dst host count	Count of connections having the same destination host
33	dst host srv count	Count of connections having the same destination host and using the same service
34	dst host same srv rate	% of connections having the same destination host and using the same service
35	dst host diff srv rate	% of different services on the current host
36	dst host same src port rate	% of connections to the current host having the same src port
37	dst host srv diff host rate	% of connections to the same service coming from different hosts
38	dst host serror rate	% of connections to the current host that have an S0 error
39	dst host srv serror rate	% of connections to the current host and specified service that have an S0 error
40	dst host rerror rate	% of connections to the current host that have an RST error
41	dst host srv rerror rate	% of connections to the current host and specified service that have an RST error

Only 10% of the data from the original KDD Cup99 Dataset is classified into different attack types, and the numbers are presented in Table 2. The KDD Cup 99 Dataset is not perfect and characterized by the presence of redundant and duplicate data sets. The details of the corrected data and original data are presented in Table 3[5].

Table 2: Attack types from KDD Cup 99 Dataset

Normal	Probing	DOS	R2L	U2R
Normal (97277)	Nmap (231)	Land (21)	Spy(2)	Buffer_overflow(30)
	Portsweep (1040)	Pod (264)	Phf(4)	Rootkit(10)
	Ipsweep (1247)	Teardrop (979)	Multihop (7)	Loadmodule (9)
	Satan (1589)	Back (2203)	ftp_write (8)	Perl(3)
		Neptune (107201)	Imap(12)	
		Smurf (280790)	Warezmaster(20)	
			Guess_passwd (53)	

Table 3: Number of attacks in training KDDCUP99 Dataset

Data Set	Normal	Probing	Dos	R2L	U2R
10%KDD	97277	4107	391458	1126	52
Corrected KDD	60593	4106	229853	11347	70
Whole	972780	41102	3883370	1126	50

The project undertook works on the above KDD dataset. A brief explanation of the various types of attack classification used in the project is listed below.

1. DoS: Stands for “Denial of Service”. The attacks prevent the legitimate user from accessing the service.
2. R2L: Stands for "Remote to Local". Victim’s machine is accessed through attacks, even though the attacker doesn't have access to the victim's machine [9].
3. U2R: Stands for “User to Root”. The attacks try to gain superuser privileges when the attacker secures the local access to a victim's machine [10].
4. Probe: The attacks try to get the information regarding the local host.

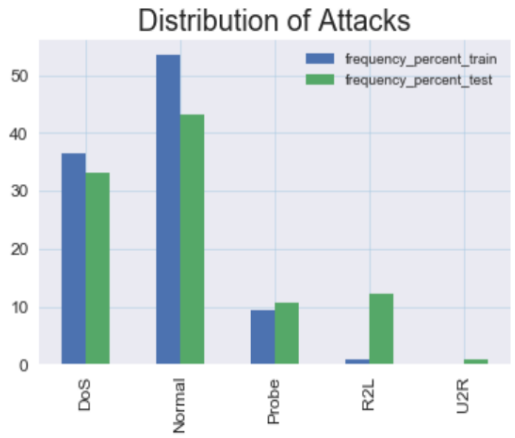


Figure1: Distribution of attacks in train and test data sets.

The project does not depend on all the parameters present in the above KDD Cup 99 Dataset. The project maps the covariance of different parameters and takes into consideration only 10 important parameters that get analyzed using machine learning algorithm and feature engineering which are detailed below.

```
In [43]: selected_final_features
Out[43]: ['src_bytes',
          'dst_bytes',
          'logged_in',
          'count',
          'srv_count',
          'dst_host_srv_count',
          'dst_host_diff_srv_rate',
          'dst_host_same_src_port_rate',
          'dst_host_serror_rate',
          'service']
```

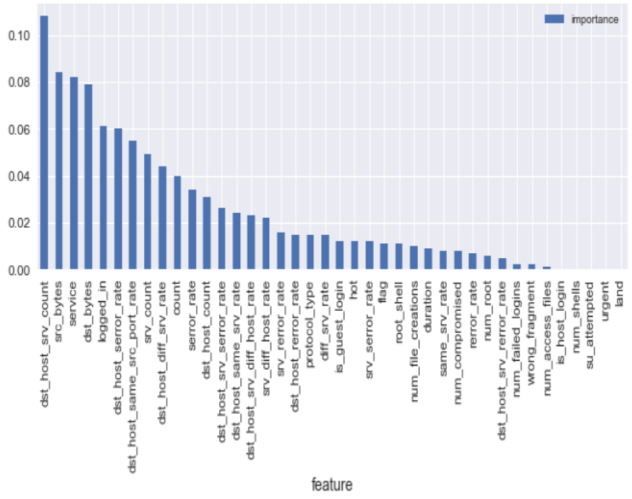


Figure2: Feature Distribution according to the level of importance.

2 DATA CLEANING

The dataset which we are using in this project has many redundancies and is too large to analyze. These redundancies lead to bias towards more frequent records. Also, every machine learning algorithm gives us the similar result as they are biased towards more frequent records and hence, we cannot distinguish the accuracies of various algorithms or classifiers hence the data needs to be cleaned. The data cleaning procedure involves the removal of duplicate or irrelevant observations, fixing structural errors, filtering unwanted outliers and handling missing data.

3 FEATURE ENGINEERING

Machine learning algorithms can be granted features and made to work using the domain knowledge of the data. This process is called feature engineering. The predictive power of machine learning algorithms can be improved by creating features from raw data that help facilitate the machine learning process[13]. By selecting only useful features, Feature engineering reduces the amount of data and avoids the meaningless calculations on the useless features. It improves accuracy by removing misleading/unwanted features and avoids the possibility of overfitting by removing the correlated features.

3.1 METHODS

We find the useless features by calculating the variance of values for each feature. Also, find correlated features by calculating the correlation coefficients between features and then normalize the features selected.

4 MACHINE LEARNING CLASSIFIERS

The class of the various data points in the dataset can be classified using the various machine learning classifiers. This classification predictive modeling is the task of approximating a mapping function (f) from input variables (x) to discrete output variables(y). In our project, we have used five different classifiers to model our intrusion detection system. They are as below.

3.1 DECISION TREE

Decision tree builds classification or regression models in the form of a tree structure [6]. The algorithm uses an if-then rule which aids in learning using the data subsets which are incrementally arranged in the form of a tree node. The rules get learned sequentially using the training data one at a time. Attributes in the top of the tree have more impact towards in the classification and are identified using the information gain concept

3.2 NAÏVE BAYES

The Naïve Bayes classifier is based upon Bayes theorem under a simple assumption which is, the attributes are conditionally independent. Even though the features are independent, at the end all of these contribute to probability. It is a simple algorithm and can be scalable to larger datasets as it takes linear time.

3.3 K-NEAREST NEIGHBOR(KNN)

KNN is a lazy learning algorithm. The algorithm stores all the instances regarding the training data points in n-dimensional space. For real-values data, this classifier returns the mean of k nearest neighbors.

3.4 RANDOM FOREST CLASSIFIER

Random forest classifier originates a group of decision trees from an irregularly collected subset of the training dataset. It then aggregates the votes from all the different tree classifiers to decide the final class of the test object.

3.5 LOGISTIC REGRESSION

A logistic Regression model uses one or more independent variables to determine the outcome. The model measures the covariance between the categorical dependent variable and one or more independent variable by estimating probabilities using a sigmoid function. The output is produced based on the weighted sum of input parameters.

5 OVER VIEW OF OUR MODEL / WORK DONE

The methodology used in our project can be summarized in five steps as shown in Fig2.

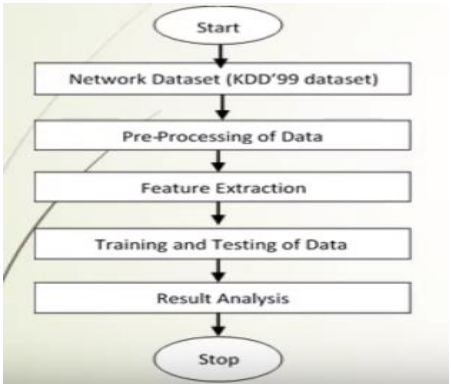


Figure 2: Work Done .

After the KDD data set is pre-processed and feature extraction is completed. A significance value is given to each feature after the feature extraction is completed. The ten of the most important features required for our model is given to the classifiers discussed in Section 4 to train and test the data and to predict the results. As the number of features is reduced, the time required in training our model is reduced.

6 ACCURACY METRICS:

Accuracy: The accuracy parameter is used to determine the performance of our model. A confusion matrix is generated which helps in the calculation of accuracy. In the matrix, if diagonal values are high, the accuracy of the classifier is high else vice versa. Accuracy is calculated using the below formula

$$Accuracy = \frac{\sum_{i=1}^{n} c_{ii}}{\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}}$$

Where n is the total number of [7] classes and Cij of a confusion matrix which is of size nxn. The matrix represents the number of class i predicted in class j.

Precision: Precision Score is obtained by dividing number of true positives with sum of number of true positives and false positives. Where false positives can be obatained by adding all the values the respective column except the true positive. When the value of presision is one then that is the good value. Zero is bad value

Recall : Recall is otherwise called as sensitivity. The value of recall is obtained by dividing true positives with the sum of true positives and false negatives. Where the sum of true positives and false negatives are the result of total test examples of a particular class.

F1 score: F1 score which is also known as specificity is obtained by dividing true negatives with the sum of true negatives and false positives. Where true negatives.

Higher values of presision results in low values of reacall. F1 score which can also be termed as specificity is measured to compare Precision and Recall

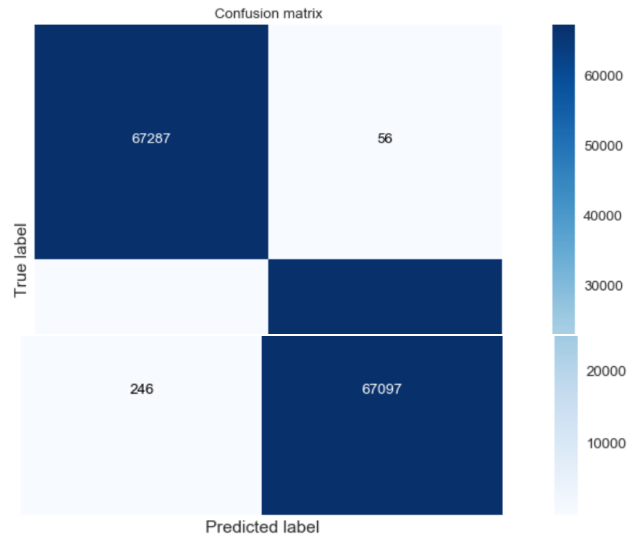
Confusion Matrix: It gives the values of true positives, true negatives, false positives and false negatives. Confusion matrix is very useful to estimate the actual performance of the machine learning model.

6 RESULTS & CONCLUSION

MODEL	TRAIN ACCURACY	TEST ACCURACY
DECISION TREE	99%	82%
NAÏVE BAYES	94%	86%
KNN	99%	89%
RANDOM FOREST	99%	85%
LOGISTIC REG	96%	85%

Since the training and texting accuracy of KNN classifier is good when compared to other classifiers. KNN classifier is taken to pickle the final model fitting and is used for preparing the intrusion detection application.

CONFUSION MATRIX OF KNN CLASSIFIER:



True positives, True Negatives, False positives, False Negatives can be seen clearly though this confusion matrix.

INTRUSION DETECTION APPLICATION

An Intrusion detection application is developed from the knn model, flak front end, HTML, CSS and java script. So, when a new file which contains the details of a network is given to the intrusion detection application, it will be able to detect weather the network is malicious or normal.

ACKNOWLEDGMENTS

We are deeply indebted to Professor Gokay Saldamli for his invaluable lectures and assistance in Network Security which is very helpful in the preparation of this study.

REFERENCES

[1] A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection, Anna L. Buczak ; Erhan Guven, DOI: 10.1109/COMST.2015.2494502.

[2]https://xplqa30.ieee.org/document/7307098?reason=concurrenc

[3] http://archive.ics.uci.edu/ml/databases/kddcup99/task.html

[4] Efficient Classifier for R2L and U2R Attacks P. Gifty Jeya , M. Ravichandran,C. S. Ravichandran, International Journal of Computer Applications (0975 – 8887), Volume 45– No.21, May 2012.

[5] Denial-of-Service, Probing & Remote to User (R2L) Attack Detection using Genetic Algorithm, Swati Paliwal, Ravindra Gupta, International Journal of Computer Applications (975 – 8887) Volume 60– No.19, December 2012

[6] https://medium.com/@sifium/machine-learning-types-of-classification-9497bd4f2e14

[7] https://link.springer.com/book/10.1007%2F978-3-319-98678-

[8] Artificial Immune Networks Based Radial Basic Function Neural Networks Construction Algorithm and Application, Jiang Zhong, Yong Feng, Chunxiao Ye,Ling Ou, Zhiguo Li. DOI: 10.1109/ICNC.2007.269

[9] Exploring wireless device driver vulnerabilities,Victor Agapov,SyedM.RahmanDOI:10.1109/ICCITECHN.2008.480313

[10] http://uvilfeed.com/PetrotechProceedings/Petrotech-2016-Proceedings/DIGITAL/Day2_6_Project-Management/A-2098.pdf Penetration and countermeasures of cyber-attacks against digital assets of an organization: Santosh Kumar Sahu1, M.K. Mathur1, C. Kumar1and Sanjay Kumar Jena2

[11] http://it.dadeschools.net/pdffiles/technologyplan2005.pdf

[12] www.slideshare.net

[13] https://letslearnai.com