Available Online at www.ijcsmc.com/journal

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC

IJCSMC, Vol. 11, Issue. 6, June 2022, pg.152 – 168

A Result Analysis of Supervised Machine Learning Approach to Detect Anomaly from Network Traffic

[1]Sharad Laxman Pawar; [2]Dr. Tryambak Hiwarkar
[1]Ph.D Research Scholar; [2] Professor
[1,2]School of Engineering and Technology
[1,2]Sardar Patel University Balaghat

DOI: https://doi.org/10.47760/ijcsmc.2022.v11i06.012

Abstract:
Supervised Machine Learning (SML) is the quest for algorithms that reason from externally given cases to develop general hypotheses, which subsequently make predictions about future instances. Supervised categorization is one of the jobs most commonly carried out by the intelligent systems. This article presents numerous Supervised Machine Learning (ML) classification strategies, evaluates various supervised learning algorithms as well as finds the most effective classification algorithm depending on the data set, the number of instances and variables (features) (features). Seven alternative machine learning methods were considered: Decision Table, Random Forest (RF) , Naïve Bayes (NB) , Support Vector Machine (SVM), utilizing Waikato Environment for Knowledge Analysis (WEKA)machine learning program. To develop the algorithms, Diabetes data set was utilized for the classification with 786 cases with eight attributes as independent variable and one as dependent variable for the analysis. The findings suggest that SVM was determined to be the method with maximum precision and accuracy. Naïve Bayes and Random Forest classification algorithms were shown to be the next accurate after SVM appropriately. The research demonstrates that time spent to create a model and precision (accuracy) is a factor on one hand; while kappa statistic and Mean Absolute Error (MAE) is another element on the other side. Therefore, ML techniques demands precision, accuracy and least error to have supervised predictive machine learning.

I. INTRODUCTION

Machine learning is one of the fastest growing areas of computer science, with far-reaching applications. It refers to the automated detection of meaningful patterns in data. Machine learning tools are concerned with endowing programs with the ability to learn and adapt [19]. Machine Learning has become one of the mainstays of Information Technology and with that, a rather central, albeit usually hidden, part of our life. With the ever increasing amounts of data becoming available there is a good reason to believe that smart data analysis will become even more pervasive as a necessary ingredient for technological progress. There are several applications for Machine Learning (ML), the most significant of which is data mining. People are often prone to making mistakes during analyses or, possibly, when trying to establish relationships between multiple features [9]. Data Mining and Machine Learning are Siamese twins from which several insights can be derived through proper learning algorithms. There has been tremendous progress in data mining and machine learning as a result of evolution of smart and Nano technology which brought about curiosity in finding hidden patterns in data to derive value. The fusion of statistics, machine learning, information theory, and computing has created a solid science, with a firm mathematical base, and with very powerful tools. Machine learning algorithms are organized into a taxonomy based on the desired outcome of the algorithm. Supervised learning generates a function that maps inputs to desired outputs. Unprecedented data generation has made machine learning techniques become sophisticated from time to time. This has called for utilization for several algorithms for both supervised and unsupervised machine learning. Supervised learning is fairly common in classification problems because the goal is often to get the computer to learn a classification system that we have created [21]. ML is perfectly intended for accomplishing the accessibility hidden within Big Data. ML hand over's on the guarantee of extracting importance from big and distinct data sources through outlying less dependence scheduled on individual track as it is data determined and spurts at machine scale. Machine learning is fine suitable towards the intricacy of handling through dissimilar data origin and the vast range of variables as well as amount of data concerned where ML prospers on increasing datasets. The extra data supply into a ML structure, the more it be able to be trained and concern the consequences to superior value of insights. At the liberty from the confines of individual level thought and study, ML is clever to find out and show the patterns hidden in the data [15]. One standard formulation of the supervised learning task is the classification problem:

The learner is required to learn (to approximate the behavior of) a function which maps a vector into one of several classes by looking at several input/output examples of the function. Inductive machine learning is the process of learning a set of rules from instances (examples in a training set), or more generally speaking, creating a classifier that can be used to generalize from new instances.

II. REVIEW METHODOLOGY

In order to perform this survey, journal articles and conference proceedings related to the IDSs, particularly machine learning-based anomaly detection systems as well as those which match the scope of this survey, were compiled. In the present survey, we followed a review direction according to the proposed conventional architecture of a machine learning-based network anomaly detection system which is depicted in Figure 1. A structured review methodology was applied in order to scrutinize the research studies on the main phases of machine learning-based anomaly detection systems as follows.

Firstly, this survey starts with a background of intrusion detection systems including the methods, properties, and multiple applications of anomaly detection systems in order to take a closer look at the fundamental concepts of this subject matter. In addition, contemporary malicious behaviors are described to simplify the meaning of abnormal behaviors in a massive amount of normal behaviors. Secondly, research articles are investigated from a pre-processing point of view to discuss how the features and network data were collected and extracted; this step helps to assess the quality of the pre-processing phase. Afterwards, as a critical module in anomaly detection systems, every research article was discussed from a machine learning perspective to categorize the research studies into four main groups of machine learning techniques which were supervised learning, unsupervised learning, deep learning, and ensemble learning techniques. Regarding the learning approach of every discussed article, we tried to discover their anomaly detection and/or classification methodologies to determine a summarized version of their learning logic. Moreover, we mentioned the gap and limitations together, accordingly proposing a future direction to address it. Meanwhile, the challenges for each category of machine learning techniques were highlighted at the end of the related sections. Furthermore, we reflected all the aspects and properties of anomaly detection techniques in the corresponding comparative tables, separately and in overall.

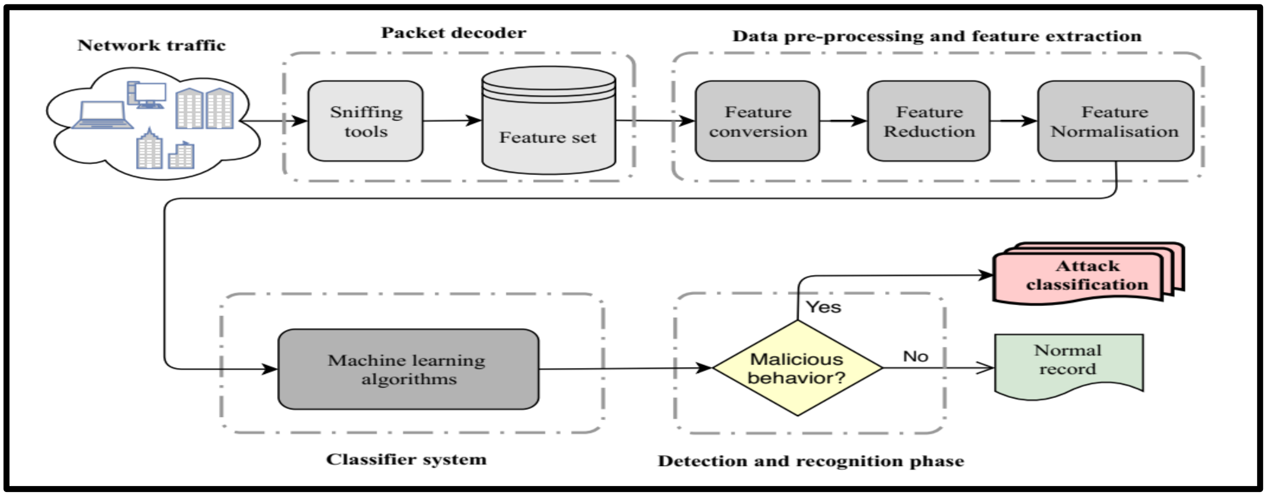


Figure 1. Main modules in machine learning classifier systems.

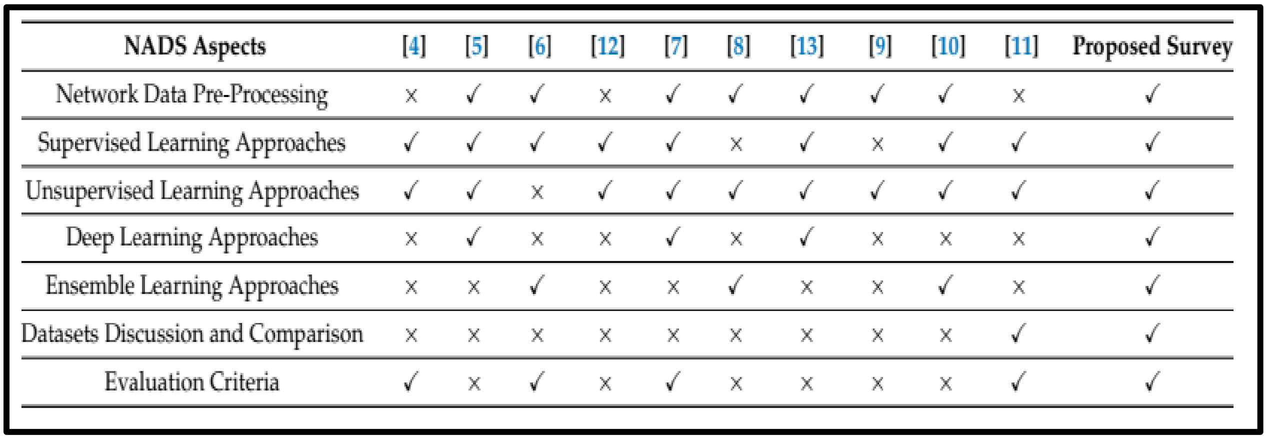
In this study, the authors focused on the main parts and properties of network anomaly detection systems, in particular, the detection phase. Numerous machine learning methods for network malicious behavior detection have been discussed towards designing an intelligent NADS which is capable of detecting known and zero-day attacks in high-speed network traffic.

2.1. Related Surveys

Over the last decade, there have been various survey articles in the literature which aimed at reviewing different types of IDSs and NADSs with varied objectives. Consequently, the present article is oriented towards the main findings of some recently published surveys in the field of IDSs and particularly anomaly detection systems; in other words, it deals with the aforementioned previous findings' merits and drawbacks. In addition, Table 1 comparatively shows the novelty and superiority of a proposed survey against recently published survey articles in the area of anomaly detection systems. Hindy et al. [4] proposed a survey about different IDS techniques and also network threats. The survey mainly focused on different types of IDSs, datasets and threats, but the authors did not cover different types of detection techniques, particularly machine learning approaches. In another study, Lu et al. [5] conducted a survey about deep learning techniques for malware intrusion detection and prediction. Although the authors provided deep learning techniques in IDSs and briefly mentioned other malware classification methods, the authors did not discuss a pre-processing step—which is an important phase in IDSs and can negatively affect the overall time complexity of a detection system. Moreover, the survey lacks a detailed classification about the variety of recently used shallow learning approaches in IDSs.

Igino et al. [6] provided a survey on adversarial attacks against IDSs on safety-critical environments. The authors provided a general taxonomy of attack tactics against IDSs and divided the detection strategy into three different phases: measurement, classification, and responses. The main challenging issue found in the detection engine in IDS techniques is implementing non-learning-based techniques to deal with the complexity of contemporary intrusions and malicious instances. Hodo et al. [7] discussed shallow and deep networks for IDSs, and the authors provided an overview of the general classification of IDSs and taxonomy with recent and past works. By comparing different learning-based techniques, they justified that the convolutional neural network (CNN) has not been exploited in the field of intrusion detection, however, they have proved that it is a good classifier. In addition, signature-based techniques are used commercially, however, the main drawback of these techniques is that they fail to detect all types of malicious instances due to not having their signature list in database. Aburomman et al. [8] conducted a survey on IDSs using ensemble and hybrid learning systems. The authors highlighted two main categories of multiple-classifier systems, homogeneous ensembles (single classification approach) and heterogeneous ensembles (two or more different classification approaches).

They proved that heterogeneous approaches based on weighted majority voting are rarely implemented for IDSs and meta-heuristic optimization techniques deserve more attention based on extracted patterns in NSL-KDD dataset. Ahmed et al. [9] provided a survey of anomaly detection techniques in the financial domain, the survey mainly focused on clustering as an unsupervised learning technique to detect fraud and anomalous data against normal ones. Different types of financial fraudulent activities such as break-in fraud, billing fraud, illegal redistribution fraud, failed logins, and the issue of the scarcity of real data have been discussed throughout their article. Buczak et al. [10] proposed a survey of data mining and machine learning techniques for cybersecurity IDSs. Both misuse and anomaly detection techniques have been discussed based on important criteria such as accuracy, complexity, time for classifying an unknown instance with a trained model, and the understandability of the final solution. The biggest gap that the authors observed was the availability of labeled data, which is a very important issue when the anomaly detection and recognition phase is based on supervised learning techniques. Ahmed et al. [11] provided a survey about network anomaly detection techniques, which focused on the few categories of detection techniques including few classification methods, statistical techniques with non-learning based techniques and few clustering approaches; moreover, the few dataset used for network anomaly detection was discussed as well, and the gap in this survey article was that it did not provide details about the pre-processing and feature extraction phases, which are very important in NADSs.



NADS Aspects	[4]	[5]	[6]	[12]	[7]	[8]	[13]	[9]	[10]	[11]	Proposed Survey
Network Data Pre-Processing	×	✓	✓	×	✓	✓	✓	✓	✓	×	✓
Supervised Learning Approaches	✓	✓	✓	✓	✓	×	✓	×	✓	✓	✓
Unsupervised Learning Approaches	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓
Deep Learning Approaches	×	✓	×	×	✓	×	✓	×	×	×	✓
Ensemble Learning Approaches	×	×	✓	×	×	✓	×	×	✓	×	✓
Datasets Discussion and Comparison	×	×	×	×	×	×	×	×	×	✓	✓
Evaluation Criteria	✓	×	✓	×	✓	×	×	×	×	✓	✓

The main contributions of this survey are highlighted as follows:

• A systematic architecture for network anomaly detection and recognition systems is proposed from a user's behavior point of view followed by the properties of intrusion detection systems and applications.

• The recent network data pre-processing tools for feature extraction comprising of feature creation, reduction, conversion and normalization are discussed.

• A comprehensive discussion on various shallow and deep learning techniques, such as supervised, unsupervised, new ensemble, and deep learning approaches are discussed followed by the challenges of designing an efficacious NADS.

• A detailed discussion of evaluation criteria, including evaluation metrics and several contemporary datasets applicable for NADSs, is provided.

III. Expected Result and Algorithm used:

3.1 K-Nearest-Neighbours(k-NN)

Because of the ease with which it may be implemented and because of the simplicity with which it functions, the k-NN technique is one that sees widespread use in the field of classification. So long as there is no break in any of the values and the NA standards are either eliminated or converted into other morals. This is true for the greatest mainstream of the procedures that make their predictions via the use of numerical computations; nevertheless, there are those that rely on text for design sympathy. However, in together of the procedures that are existence used in this scheme, there are sure issues or numeric standards that need to be present in the informations set for the procedure to function in the correct manner. When making use of an algorithm, it is often the case, or in the vast majority of instances, that it is necessary to provide the real prototypical that is going to be generated with one or more parameters.

3.2 Support Vector Machine

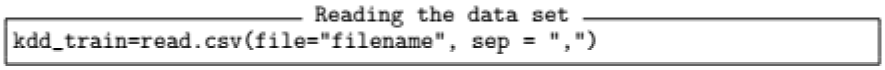
When it comes to classification, SVM is another method that sees a lot of usage. This is mostly owing to the fact that it has a high level of accuracy and that it can be used to the process of multiclass classification. The support vector machine (SVM) is the method that is utilized as the classification algorithm in the many articles that have been published regarding anomaly detection via machine learning. Many of the studies may demonstrate accuracy in the numbers that is close to one hundred percent, which is the desired end but is almost hard to attain at least in a real world situation. This is the wanted outcome. In the actual world, where there are so many assaults, and where the dangers continually keep evolving, making it tougher to see them. In order for machine learning to successfully replace traditional IDS, a significant amount of training data is necessary. Simply being able to discuss the numerical results that a large number of the researchers have obtained in the course of their experiments is a significant accomplishment that represents a significant step forward for machine learning.

IV. Preparation of the data set

Earlier the informations set can be utilized by the procedures, there are a few specific things that need to be done, as was previously stated in the methodology. Because of the data collection, a number of different conversions and adjustments will need to be carried out before the algorithms can be run successfully and without committing any errors.

4.1 Delivered informations set with terms on structures

The firstly object that was complete with the informations set, was recite it into R studio in instruction to understand the entire informations set and to usage it later. This is effortlessly complete by this streak:



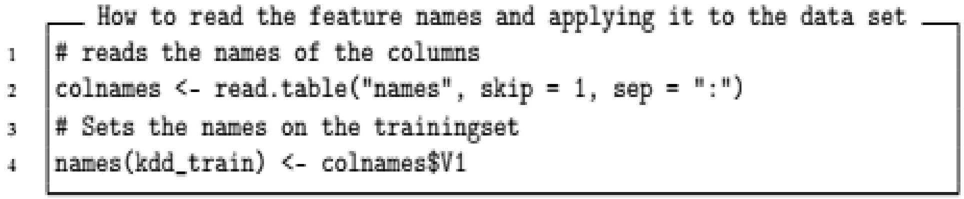
Reading the data set

```
kdd_train=read.csv(file="filename", sep = ",")
```

This particular line of code is responsible for reading the data set and saving it under the name kdd train. When interpretation the informations customary into an informations edge, which is what it's called in R, it separates the informations into columns or topographies by using "," as a separator between each column and feature.

The next step is to assign labels to the various columns that were produced as a result of understanding the informations established. This is accomplished by reading a file that has all of the article designations and then smearing it to the data set. This ensures that the rows have the right names and that it is simple to call upon those rows when using and updating the numerous topographies.

This is not an overly difficult task to do, but it is essential to ensure that the data collection you are working with is clean and accurate. In the script that is being utilised for this project, the process was carried out as follows:



How to read the feature names and applying it to the data set

```
# reads the names of the columns
colnames <- read.table("names", skip = 1, sep = ":")
# Sets the names on the trainingset
names(kdd_train) <- colnames$V1
```

4.2 SVM Binary Classification Result:

The initially procedure that was utilized in the studies was the SVM algorithm, and it consumed great informations when looking solely at the correctness of the forecast that was complete founded on the perfect of the technique. The actual assaults and the forecast were quite similar to one another, as can be seen in figure 5.1.

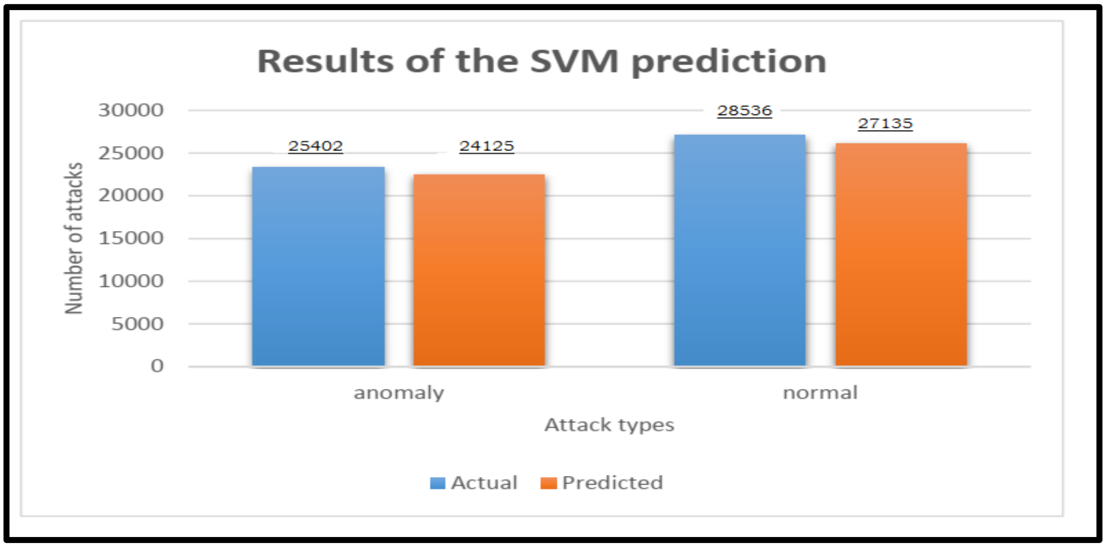


Figure 2 This bar chart illustrates the change in the amount of forecasts that were properly categorized using SVM binary classification and the total number of predictions. Additionally, it shows how many genuine assaults are there in the set of data and how those it should've been able to spot.

In figure 5.1, the actual amount of different sorts of attacks or regular circulation taken from the informations set is signified by the bars situated to the left in both of the categories. This number can be found by looking at the Actual column. The forecast demonstrated a high level of accuracy in both the normal and abnormal scenarios, with less than one thousand wrong classifications overall. When the data set contains around 50,000 items, it is not a terrible outcome to have just under 2,000 of those items wrongly categorized. The total accuracy of the system is 96.5%, as determined by the computation of how many percent are identified. That determines that the error rate is 3.5 percent if the accuracy is 96.5%, which it would be if the accuracy was 100%. It is not quite perfect, but it is really near, and that is a satisfactory level of performance.

4.3 k-NN Binary Arrangement

In order to carry out binary classification using the data set containing binary information, the next approach that was used was called K-NN, which stands for k-Nearest Neighbors. This algorithm also shown remarkable results, achieving high percentages of accuracy in its calculations. This is evident from the bar chart that is shown in figure 5.2.

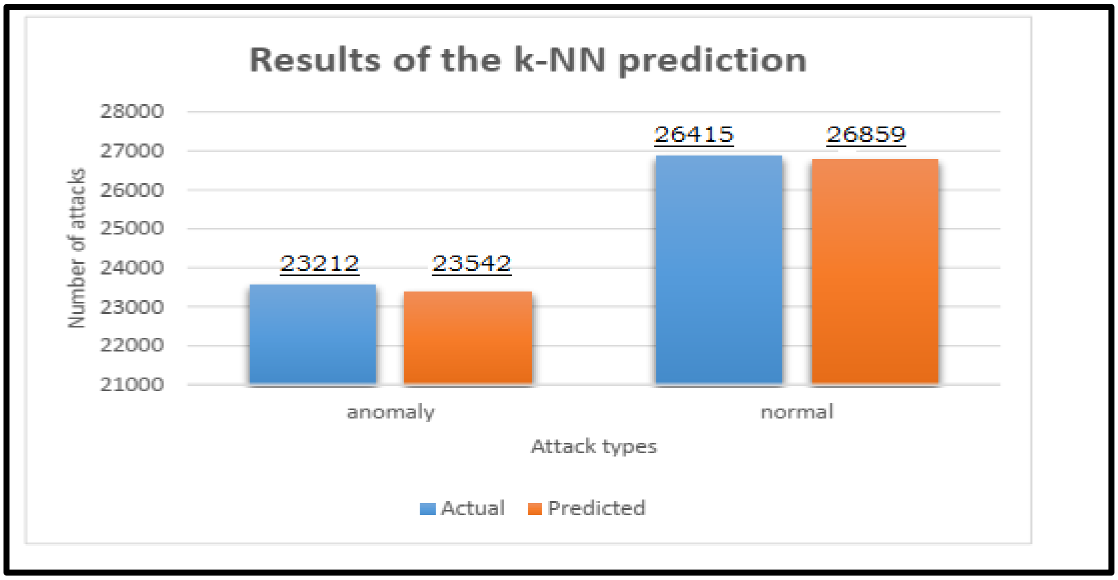


Figure 3: This bar chart illustrates the alteration among the number of forecasts that were successfully categorized using k-NN binary classification and the entire amount of forecasts. It also indicates the real number of distinct sorts of assaults that should have been identified by the system and how many there are in the data set.

This adheres to the same presentation rules as figure 5.1, which makes use of SVM; it has the same labels, and its distinct classes are described as abnormal and normal, respectively. The bar that is shown to the left of the grouping represents the real amount of bouts that are included in the data set. This bar is shown. The one on the right is the predictions that were generated by the script using the enhanced limitations as contribution, and it is based on the model on the left.

4.4 SVM Multiclass Classification

The first research that is carried out in experiment two is the one in which multiclass classification is carried out using the SVM method. This will be accomplished via the use of supervised learning, a subclass of machine learning that was discussed in detail in the previous episode's background section. This is what is used in each of the first two tests that are carried out. Specifically, the trials in which the algorithms are put to use, as opposed to the monitoring of the amount of time and resources used. In a nutshell, what this supervised learning entails is that the algorithms that are employed, while being trained, already know whatever the bout is, and they learn by combining the various assaults and recognizing the designs after the group of bouts. This is how the supervised process works. Therefore, in a nutshell, it already knows what to look for since it has studied the patterns of the classes in advance and is now searching for those patterns inside the test data set.

In this experiment, all of the aforementioned things are done, and it will employ machine learning methods (also known as supervised learning) in conjunction with the algorithm SVM. As was discussed previously in the background chapter, the support vectors and classifiers that are used by the SVM method are hyper planes. And the technique did not have the finest presentation in this test when observing at the correctness statistics in figure 5.3, which presents the presentation of SVM while utilizing multiclass arrangement. It is clear from looking at the bar chart that this experiment is very distinct from the one that came before it. The reason for this is because it was classified using more than one class. In the chart, there are a total of five 10 bars, with two bars devoted to each assault.

The first of the two is the one that indicates in what way numerous in total there are in the data set pertaining to that particular assault. The second bar presents information on the total number of incidents that may be attributed to a certain kind of assault. The bars at R2L and U2R are not missing; nonetheless, there are far fewer of them than there are of the other types.

There ought to be three R2L, but it only recognizes one of them as such, and there ought to be thirteen U2R, but it only recognizes eight of them as such. And when observing at the amount that should be of R2L, that is, when likened to the amount of usual circulation, it becomes obvious that the bars will wind up being little on the ones that have such little statistics.

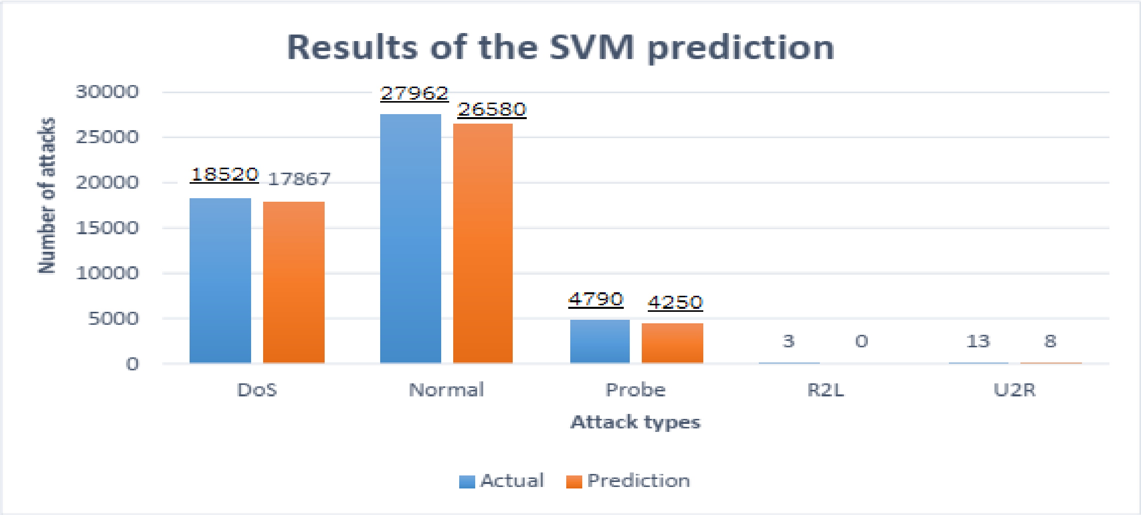
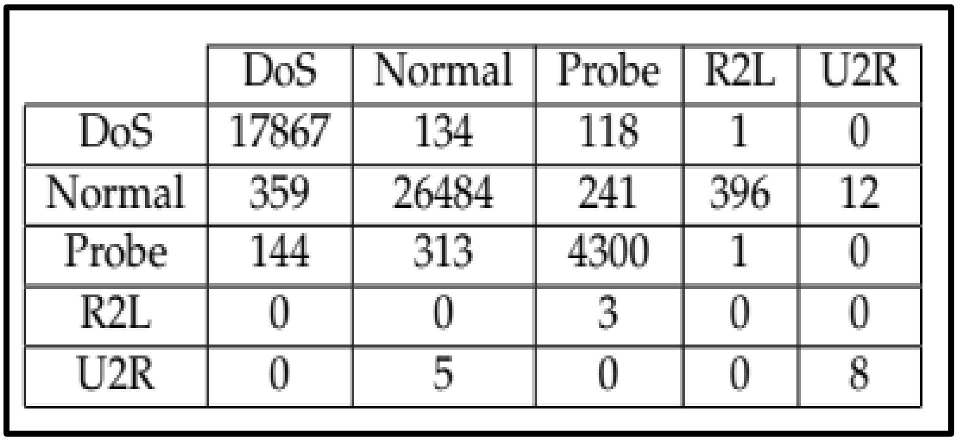


Figure 4: ***This bar chart illustrates the alteration among the number of forecasts that were successfully categorized using SVM multiclass classification and the entire amount of predictions. It also indicates the real number of distinct sorts of assaults that should have been identified by the system and how many there are in the data set.***

In the commencement of this part recitation the trial, I said that there are several additional aspects that may consider in on the correctness or presentation of the procedure.

There are a few different aspects to consider, such as the dividing of the informations group. There are varying amounts of examples from each of the distinct assaults, and since they are categorized into just five groups rather than 22, there is a possibility that some errors were made as a result of this. The data set that comprises 22 distinct assaults includes specific attacks, and those attacks may have varying values depending on which columns are utilized. Therefore, smooth however there must be approximately in the exercise usual, they strength be misidentified as normal traffic because the model might not recognize the specific attack because it might not have ever seen that exact design beforehand. This could be the case even though there should be some in the training set. When conducting the classification in the way that it is done in this project, this bottleneck presents a challenge that is difficult to overcome.



	DoS	Normal	Probe	R2L	U2R
DoS	17867	134	118	1	0
Normal	359	26484	241	396	12
Probe	144	313	4300	1	0
R2L	0	0	3	0	0
U2R	0	5	0	0	8

Table 5.6: The misperception matrix for SVM multiclass arrangement

The information that is shown in table 5.6 pertains to the number of samples that were successfully and wrongly categorized by the algorithm. The samples that were successfully identified are shown in the columns that correspond to them thanks to the alignment of the columns. Therefore, the accurately predicted DoS samples can be found in the column labelled "DoS," which is the very first column in both the straight and perpendicular directions and is located far to the left in the table. The column in question is the one that has the number 17867. This is the column that is being discussed. The remaining ones in the similar row straight are the ones that must have stood categorized as DoS but were not.

4.5 k-NN Multiclass Classification

In order to do multiclass classification with k-NN, it is necessary to make a great number of adjustments that are comparable in kind. As was discussed before in this thesis, K-NN operates in a manner that is distinct from that of SVM. It determines the result of the following categorization based on which K-Nearest Neighbors' votes get a majority of the total votes cast. The Closest Neighbor algorithm serves as the foundation for this approach; however, rather of focusing on the single nearest neighbor, this algorithm considers all k of those neighbors instead.

K may have any value, and there are a lot of different ideas floating around about what value it should have. Some people believe that the value of k must be anywhere close to the square root of the number of admissions or examples in the informations usual, while others claim that they utilized the value of k that was optimized. In order to get the greatest results with k in this project, the tune.knn() function was used. This is a piece of software that calculates and returns the optimal

4.6 Research Result three:

This experiment will also include a measurement of how much time was used, which is another extremely significant consideration. There is nothing novel about the concept that time is of paramount significance; anything new ought to function quickly, and if it does not function quickly, it ought to produce extraordinary results. The process of retrieving a period imprint from the R server is what is done in order to determine how quickly the various scripts and algorithms execute. This step is taken both at the beginning of the script and at the end, once everything else has been completed. This knowledge or line of cryptogram is as it is indicated in the Result I: System. The time stamp is retrieved by executing a command that requests the system for its time stamp ().

This information is then saved as the start. Time and end. Time variables, after which it is compared and the amount of time that was used is shown by subtracting the end. Time value from the start. Time value. As a notice, this is what is complete in instruction to count the amount of time that has been used, although everything is detailed much more thoroughly in the chapter that came before this one.

Due to the fact that the binary analysis had fewer categories into which to place the data, it was anticipated that this method would need less time to complete than the multiclass analysis did.

Surprisingly, this did not at all turn out to be the case; when looking at the same method, the amount of time required for binary classification and multiclass classification was identical. Both the SVM and the k-NN algorithms required vastly different amounts of time to complete. The screenplays for one of them had a run length of around three and a half hours, while the scripts for the other one utilized little more than one and a half hours. The fact that one of them took a great deal more time than the other is a very important consideration. As was noted before, the data collection has around 126,000 different samples in total. When both the quantity and the amount of time are considered, it is possible to reach the conclusion that the processing of 126,000 samples in slightly more than an hour is an acceptable amount of time, but that processing the data set for three hours is an extremely lengthy amount of time. To put this into perspective in the real world, if the device being used is linked to the internet and has the ability to utilize SSH, it is possible to rapidly output 126 thousand lines in a log in a very short amount of time. Because large businesses may have over 126,000 lines in their logs after just an hour of activity, time may become a problem when attempting to run the algorithm using the specifications that were utilized for this project.

The processing of the data would need to include additional options, such as those powered by a GPU or several cores.

Conclusion:

The most well-known supervised approaches are described in this work in reasonable depth. We should point out that our list of references is not a comprehensive list of articles covering supervised techniques; our goal was to offer a critical evaluation of the essential ideas, rather than a simple list of all publications that mentioned or used those ideas. In spite of this, we hope that the references mentioned address the most important theoretical topics, and provide access to the main branches of the literature dealing with such approaches, pointing the researcher in intriguing study paths.

When it comes to ML classification, the crucial question is not whether one learning algorithm is better than another, but rather under what circumstances a certain approach might outperform others on a specific application problem. Toward this end, meta-learning is looking for functions that connect datasets to algorithm performance. To this purpose, meta-learning examines the relationships between a collection of variables known as meta-attributes and the performance of learning algorithms in order to describe the features of learning tasks. The number of examples, the fraction of categorical attributes, the percentage of missing data, the entropy of classes, etc. are some of the features of learning tasks

Once the merits and weaknesses of each approach have been analyzed, the idea of combining multiple algorithms to solve a problem should be examined. The goal is to make use of the advantages of one approach while minimizing the drawbacks of the other. A single classifier that does as well as a decent ensemble of classes may be difficult or impossible to come across, if all we care about is the greatest possible classification accuracy. There are at least three drawbacks to ensemble techniques, despite their evident advantages. Because all component classifiers must be kept, rather than just a single classifier, the first issue is that it requires more storage space. The overall amount of storage is determined by the size of the ensemble and the individual component classifiers (number of classifiers in the ensemble). Because all component classifiers (as opposed to a single classifier) must be evaluated in order to categories an input query, the second drawback is an increase in computation time. The final flaw is a lack of comprehension. Non-expert users have a harder time understanding the logic behind a choice when numerous classifiers are involved.

The use of ensemble techniques is only recommended if we are only concerned with obtaining the best possible categorization results. The wrapper feature selection approach (Guyon & Elissee, 2003) is another time-consuming attempt to improve classification accuracy without compromising understandability. More characteristics should, in theory, lead to more discriminating capacity. In practice, however, this is not necessarily the case with machine learning algorithms. Wrapper approaches employ cross-validation to forecast the advantages of adding or deleting a feature from the feature subset used in the induction procedure.

When it comes to computer science, a lot of academics are used to dealing with flat data and algorithms that can be executed in only a few minutes or seconds. For these researchers, the "extremely big" datasets begin at 100,000 occurrences with two dozen attributes. Databases, on the other hand, have to cope with gigabytes of data. However, it is improbable that all the data in a data warehouse would be mined at the same time

For many real-world situations and databases, many of the existing learning techniques are computationally costly and need storing all data in main memory. Partitioning the data is an orthogonal strategy that eliminates the requirement for algorithms to be performed on extremely big datasets. Datasets are partitioned into subsets, each of which is learned concurrently, and the results are combined (Basak and Kothari, 2004). This concurrent execution of machine learning processes may be carried out via distributed agent systems (Klusch et al., 2003). For local processes, nonparallel machine learning methods can still be deployed because they don't need to know about information from other data sources. Agents must work with other agents to incorporate data from a wide variety of local sources.

References

[1] Alex S.& Vishwanathan, S.V.N. (2008). Introduction to Machine Learning. Published by the press syndicate of the University of Cambridge, Cambridge, United Kingdom. Copyright Ⓒ Cambridge University Press 2008. ISBN: 0-521-82583-0. Available at KTH website: https://www.kth.se/social/upload/53a14887f276540ebc81aec3/online.pdf Retrieved from website: http://alex.smola.org/drafts/thebook.pdf

[2] Bishop, C. M. (1995). Neural Networks for Pattern Recognition. Clarendon Press, Oxford, England. 1995. Oxford University Press, Inc. New York, NY, USA ©1995 ISBN:0198538642 Available at: http://cs.du.edu/~mitchell/mario_books/Neural_Networks_for_Pattern_Recognition_-_Christopher_Bishop.pdf

[3] Brazdil P., Soares C. &da Costa, J. (2003). Ranking Learning Algorithms: Using IBL and Meta-Learning on Accuracy and Time Results.Machine LearningVolume 50, Issue 3,2003.Copyright ©Kluwer Academic Publishers. Manufactured in The Netherlands, doi:10.1023/A:1021713901879pp. 251–277. Available at Springer website: https://link.springer.com/content/pdf/10.1023%2FA%3A1021713901879.pdf

[4] Cheng, J., Greiner, R., Kelly, J., Bell, D.& Liu, W. (2002). Learning Bayesian networks from data: An information-theory based approach. Artificial Intelligence Volume 137. Copyright © 2002. Published by Elsevier Science B.V. All rights reserved pp. 43 – 90. Available at science Direct: http://www.sciencedirect.com/science/article/pii/S0004370202001911

[5] Domingos, P. & Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning Volume 29, pp. 103–130 Copyright © 1997 Kluwer Academic Publishers. Manufactured in The Netherlands. Available at University of Trento website: http://disi.unitn.it/~p2p/RelatedWork/Matching/domingos97optimality.pdf

[6] Elder, J. (n.d). Introduction to Machine Learning and Pattern Recognition. Available at LASSONDE University EECS Department York website: http://www.eecs.yorku.ca/course_archive/2011-12/F/4404-5327/lectures/01%20Introduction.pdf

[7] Good, I.J. (1951). Probability and the Weighing of Evidence, Philosophy Volume 26, Issue 97, 1951. Published by Charles Griffin and Company, London 1950.Copyright © The Royal Institute of Philosophy 1951,pp. 163-164.doi: https://doi.org/10.1017/S0031819100026863. Availableat Royal Institute of Philosophy website: https://www.cambridge.org/core/journals/philosophy/article/probability-and-the-weighing-of-evidence-by-goodi-j-london-charles-griffin-and-company-1950-pp-viii-119-price-16s/7D911224F3713FDCFD1451BBB2982442

[8] Hormozi, H., Hormozi, E. & Nohooji, H. R. (2012). The Classification of the Applicable Machine Learning Methods in Robot Manipulators. International Journal of Machine Learning and Computing (IJMLC), Vol. 2, No. 5, 2012 doi: 10.7763/IJMLC.2012.V2.189pp. 560 – 563. Available at IJMLC website: http://www.ijmlc.org/papers/189-C00244-001.pdf

[9] Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. Informatica 31 (2007). Pp. 249 – 268. Retrieved from IJS website: http://wen.ijs.si/ojs-2.4.3/index.php/informatica/article/download/148/140.

[10] Lemnaru C. (2012). Strategies for dealing with Real World Classification Problems, (Unpublished PhD thesis) Faculty of Computer Science and Automation, Universitatea Technica, Din Cluj-Napoca. Available at website: http://users.utcluj.ro/~cameliav/documents/TezaFinalLemnaru.pdf

[11] Logistic Regression pp. 223 – 237. Available at: https://www.stat.cmu.edu/~cshalizi/uADA/12/lectures/ch12.pdf

[12] Neocleous C. & Schizas C. (2002). Artificial Neural Network Learning: A Comparative Review. In: Vlahavas I.P., Spyropoulos C.D. (eds)Methods and Applications of Artificial Intelligence. Hellenic Conference on Artificial IntelligenceSETN 2002. Lecture Notes in Computer Science, Volume 2308. Springer, Berlin, Heidelberg, doi: 10.1007/3-540-46014-4_27 pp. 300-313. Available at: https://link.springer.com/chapter/10.1007/3-540-46014-4_27.

[13] Newsom, I. (2015). Data Analysis II: Logistic Regression. Available at: http://web.pdx.edu/~newsomj/da2/ho_logistic.pdf

[14] Nilsson, N.J. (1965). Learning machines. New York: McGraw-Hill.Published in: Journal of IEEE Transactions on Information Theory Volume 12 Issue 3, 1966. doi: 10.1109/TIT.1966.1053912 pp. 407 – 407. Available at ACM digital library website: http://dl.acm.org/citation.cfm?id=2267404

[15] Pradeep, K. R. & Naveen, N. C. (2017). A Collective Study of Machine Learning (ML) Algorithms with Big Data Analytics (BDA) for Healthcare Analytics (HcA). International Journal of Computer Trends and Technology (IJCTT) – Volume 47 Number 3, 2017. ISSN: 2231-2803, doi: 10.14445/22312803/IJCTT-V47P121, pp 149 – 155. Available from IJCTT website: http://www.ijcttjournal.org/2017/Volume47/number-3/IJCTT-V47P121.pdf