EVALUATION OF ACCURACIES AND PREDICTION TIME OF MACHINE LEARNING TECHNIQUES IN AN ANOMALY DETECTION SYSTEM

Nhat Hieu Le - 101124254 Zein Hajj-Ali - 101020677

I/ Problem Summary

It has been shown that the traditional Network Intrusion Detection System (NIDS) has imposed limitations: Zero-day exploitation, high False Alarm Rate (FAR) and unable to process encrypted packets. Recently, Machine Learning (ML) and Deep Learning (DL) techniques have been becoming promising alternative approaches to overcome those aforementioned disadvantages. Our project's goal is to investigate research papers working on NIDS using DL algorithms then we will compare the accuracies of proposed DL solutions with popular ML alternatives mentioned in [2] such as: Naive Bayes, Random Forest, Bayes Network,... Then we will compare the prediction time it will take to classify the batch of network traffic besides accuracy (assuming that our NIDS discretely analyzes a chunk of network traffic), time efficiency is also an important metric for the NIDS as malicious activities need to be predicted and be tackled as soon as possible. The dataset we will use for our evaluation purpose is UNSW-NB15 [1]. Additionally, the training time is also considered as another metric for comparison between Deep Learning methods.

II/ Project Status

Dataset Preparation

Mustafa et al. [1] generated the UNSW-NB15 at UNSW Canberra. The authors employed IXIA PerfectStorm to generate both normal and malicious traffic then Tcpdump tools are used to capture 100 GB of raw network traffic. Next, 49 features are extracted by applying Argus, Bro-IDS tools and twelve algorithms.

The full dataset contains more than 2.5 million records split into 4 CSV files and the total 49 features are grouped into 5 categories: flow, basis, content, time and additional generated features. Network traffics are not only labeled as benign and malicious, but the attack types are further splitted into 9 classes as follows: Analysis, Backdoors, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode and Worms. The partitions of the full dataset are also prepared with

175,341 records in the training set and 82,332 records in the testing set. Six features have been removed from the original dataset (“srcip”, “sport”, “dstip”, “dsport”, “stime” and “ltime”) and two new features are added (“id” and “rate”). In this project, we will utilize both a full and prepared dataset. For the full dataset, data preprocessing was performed in below manner:

- Merge four CSV files into one.
- Column labels cleaned up.
- Whitespace in string values cleaned.
- Drop duplicate records.
- Replace ‘-’ and null attributes with 0.
- Empty ‘attack_cat’ values in non-anomalous records replaced with ‘Normal’

As Deep Learning approaches work on numeric value features only, thus feature transformation needs to apply to nominal categorical features first. Label encoder is employed to perform this action. Additionally, feature scaling through standardization (Z-score normalization) is also employed within the data preprocessing process.

Learning Approach

A number of deep learning methods were chosen to be compared. The first was the multi-layered feed forward neural network proposed by Al-Zewairi, Almajali, and Awajan in [3]. After removing everything except the top 20% of features chosen by the Gedeon method, we are left with 33 features plus the label. The model has five hidden layers, with 10 neurons each for a total of 50 hidden neurons. The referenced paper found that the ReLU activation function had the lowest training time as well as convergent results. Therefore the ReLU function was used as the activation function for the hidden layers of the model. The Sigmoid function was used for the output layer since we are performing binary classification. The model was trained using 10-fold cross-validation as proposed by the paper. The model was built in Google Colaboratory using TensorFlow and Keras for the architecture and training of the model, SkLearn for the stratified k-fold cross-validation, and Pandas for dataset manipulation. When using the prepared dataset, an accuracy of 93.0% was obtained. After confirming that the model works on a smaller subset of the data, the full combined dataset containing over 2 million records was used for training and testing, resulting in an accuracy of 99.2%.

The second deep learning method to be implemented and compared used a deep autoencoder (DAE) to learn the structure and encodings of the records in an unsupervised learning environment proposed by Muna AL-Hawawreh, et.al in [5]. The DAE's architecture consisted of an encoding structure using an input layer of 42 nodes (corresponding to the number of features of the records) and a hidden layer of 10 nodes, the bottleneck layer had three nodes, and the decoding layer consisted of a hidden layer of 10 nodes and an output layer of 42 nodes. The DAE was trained on 20% of the prepared portion of the dataset, consisting of only 'normal' labeled records. The top output layer of the deep autoencoder was then removed and replaced with an output layer used for binary classification (1 node). This adjusted model was then trained on 60% of the remaining prepared records and tested on the last 20%. The resulting model achieved an accuracy of 92.2% on the test data. It is assumed that training the model on the full unprepared data will result in a higher accuracy when testing.

The goal of this project is to conduct a comprehensive comparison of different approaches using both Machine Learning and Deep Learning techniques for NIDS in terms of accuracy and execution time. To the best of our knowledge, the first proposed paper is chosen as it achieves the highest accuracy for anomaly detection. We use a deep autoencoder as the second deep learning method since it might allow for learning different encodings of the features and catch some anomalous records that might be missed in a conventional feed-forward neural network. It can also be assumed that the classification time will be quicker when using the deep autoencoder since it consists of fewer layers and nodes.

III/ Outstanding Items

There are a number of things to be done for this project before significant results can be found. The Deep Autoencoder must be trained on the full dataset consisting of over 2,000,000 records and the resulting model evaluated on a testing subset. The multi-layered feed-forward neural network will also be adjusted to become a multiclass classifier to predict the attack types as well as whether or not the record is anomalous. This might have a varied accuracy on some attack types since not all the types are equally represented in the data. The full and prepared datasets will also be used to evaluate the conventional well-known machine learning algorithms mentioned in [2] in Weka and the results compared. Ultimately, the prediction times for each of the implementations of these models will be compared.

References

1. N. Moustafa, J. Slay, UNSW-Nb15: a comprehensive data set for network intrusion detection systems (UNSW-nb15 network data set), in: Military Communications and Information Systems Conference (MilCIS), 2015, IEEE, 2015, pp. 1–6..
2. Das A., Ajila S.A., Lung CH. (2020) A Comprehensive Analysis of Accuracies of Machine Learning Algorithms for Network Intrusion Detection. In: Boumerdassi S., Renault É., Mühlethaler P. (eds) Machine Learning for Networking. MLN 2019. Lecture Notes in Computer Science, vol 12081. Springer, Cham, doi: 10.1007/978-3-030-45778-5_4.
3. M. Al-Zewairi, S. Almajali and A. Awajan, "Experimental Evaluation of a Multi-layer Feed-Forward Artificial Neural Network Classifier for Network Intrusion Detection System," *2017 International Conference on New Trends in Computing Sciences (ICTCS)*, Amman, 2017, pp. 167-172, doi: 10.1109/ICTCS.2017.29.
4. N. Moustafa and J. Slay, "The Significant Features of the UNSW-NB15 and the KDD99 Data Sets for Network Intrusion Detection Systems," *2015 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS)*, Kyoto, 2015, pp. 25-31, doi: 10.1109/BADGERS.2015.014.
5. Muna, A. H., Moustafa, N., & Sitnikova, E. (2018). Identification of malicious activities in industrial internet of things based on deep learning models. Journal of Information Security and Applications, 41, 1-11