A Robust Comparison of the KDDCup99 and NSL-KDD Intrusion Detection Datasets Through Various Machine Learning Algorithms

GEORGE MASON UNIVERSITY | College of Science

Suchet Sapre, Pouyan Ahmadi, Khondkar Islam

Thomas Jefferson High School for Science and Technology

Aspiring Scientists' Summer Internship Program

Introduction and Purpose

- With the rapid development of the internet, intrusion attacks on IoT networks have been growing exponentially.
- Millions of internet users and companies are liable to cyberattacks.
- Developing methods to identify these network intrusions is one of the most prevalent problems in cybersecurity research.
- Intrusion classification algorithms rely on data from IoT networks in order to "learn" patterns that identify compromised networks.
- Two prominent datasets used for intrusion classification: **KDDCup99** and **NSL-KDD**
- The KDDCup99 dataset was created in 1999 and has been used in many research studies, however, it has many inefficiencies.
- The NSL-KDD dataset, which is a subset of the KDDCup99 dataset, was created in response to these flaws.

Sample of the Datasets

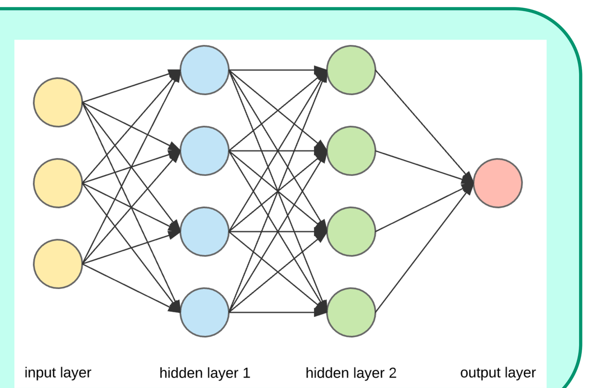
src_bytes	dst_bytes	logged_in	diff_srv_rate	…	Intrusion Type
235	1337	1	0.0	…	"normal"

- List of intrusion types: "normal", "dos", "r2l", "u2r", and "probe".
- **Purpose**: To determine the relative quality of both datasets by comparing the performance of various machine learning algorithms on both datasets with classification metrics.

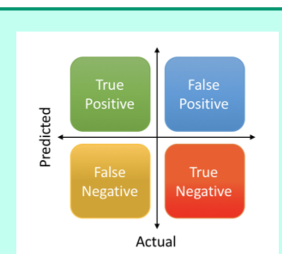
Methods

Data Preprocessing: First, the datasets were downloaded from their respective sources [1] and [2]. Then, this data had to be formatted such that a classifier could accept it as input. For example, one of the steps was converting the categorical columns in the dataset to one-hot encodings. A majority of the preprocessing was done through NumPy and Pandas.

Evaluation with Machine Learning Algorithms: I utilized four ML classifiers: Artificial Neural Networks, Support Vector Machines, Naïve Bayes, and Random Forests. This was done via Python ML packages and frameworks such as Sklearn and TensorFlow among others [5].



Interpretation of Results: With the use of classification metrics, I was able to compare the results of the ML classifiers trained on both datasets. I used Python packages such as Matplotlib and Seaborn for the visualizations [6].

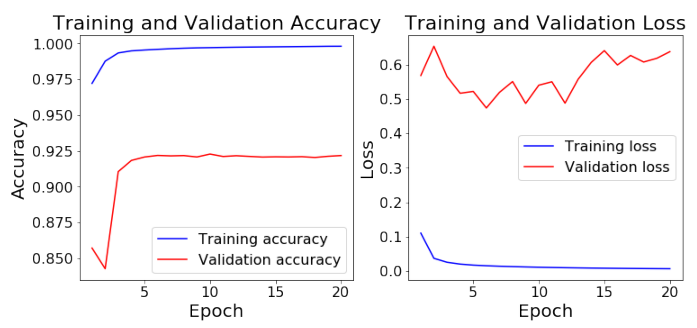


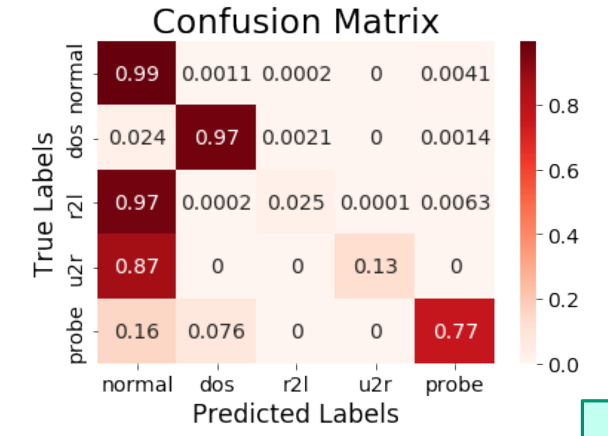
Results

KDDCup99 | **NSL-KDD**

Artificial Neural Network

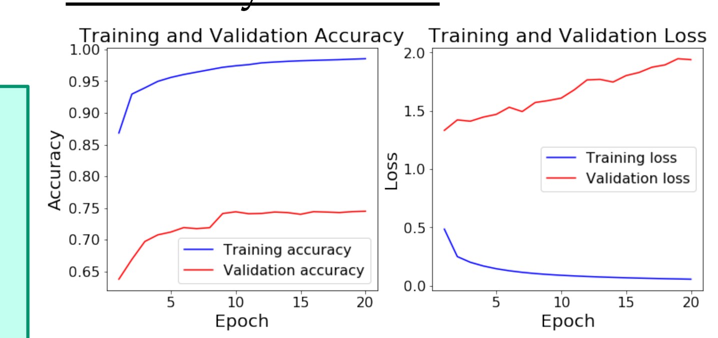
KDDCup99 — Accuracy: 92.39 %

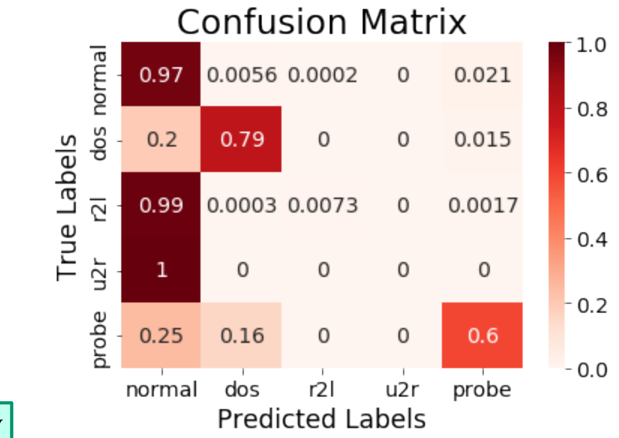




ANN Architecture
- Two hidden layers with 100 nodes
- 5 node output layer with SoftMax activation

NSL-KDD — Accuracy: 78.51%





Naïve Bayes

KDDCup99 — Accuracy: 89.94%

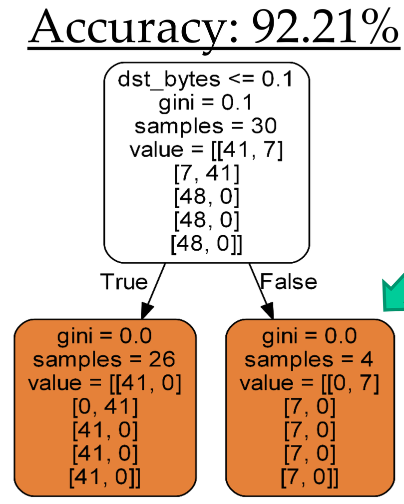
$$P(c|x) = \frac{P(x|c)\cdot P(c)}{P(c)}$$

Likelihood; Class Prior Probability; Posterior Probability; Predictor Prior Probability

NSL-KDD — Accuracy: 63.12%

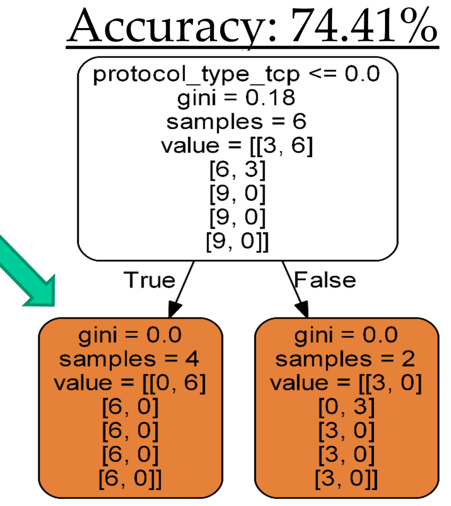
Random Forest

KDDCup99 — Accuracy: 92.21%



Example Random Forest Estimators

NSL-KDD — Accuracy: 74.41%

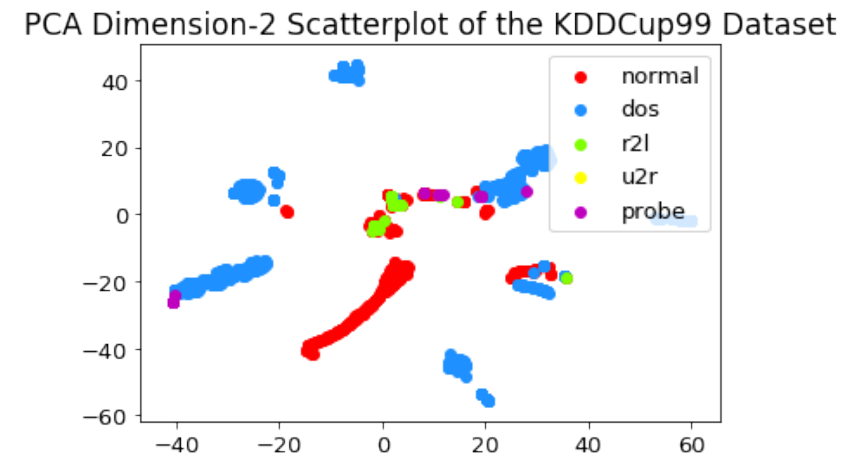


Principal Component Analysis and Classification Metrics

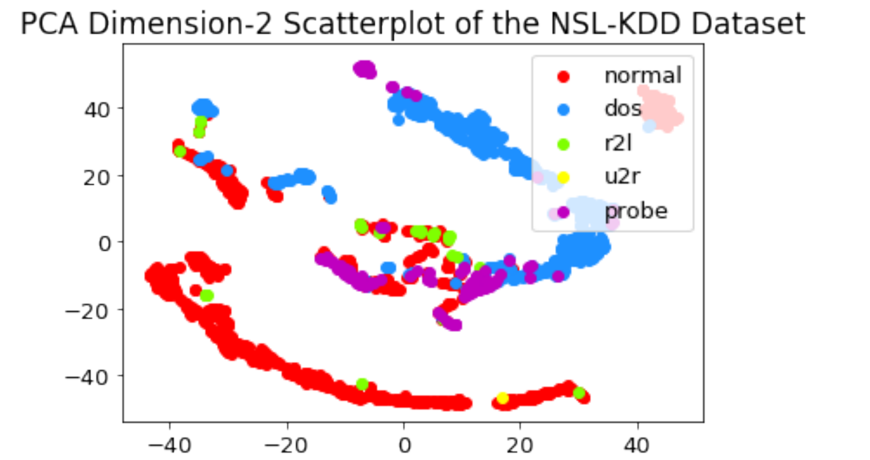
Binary Classification Metrics for each Algorithm

	KDDCup99				NSL-KDD			
	ANN	SVM	Naïve Bayes	Random Forest	ANN	SVM	Naïve Bayes	Random Forest
Precision	0.9985	0.9833	0.9937	0.9987	0.9661	0.8839	0.9672	0.9683
Recall	0.9112	0.9339	0.8537	0.9084	0.6205	0.8142	0.1746	0.6158
F1 Score	0.9529	0.9579	0.9185	0.9514	0.7557	0.8476	0.2957	0.7528

PCA Dimension-2 Scatterplot of the KDDCup99 Dataset



PCA Dimension-2 Scatterplot of the NSL-KDD Dataset



Conclusions

- The quality of the NSL-KDD dataset is significantly higher than that of the KDDCup99 dataset. This conclusion is based off the metrics used to evaluate the performance of the various Machine Learning classifiers.

Accuracy: The respective testing accuracies of the ML classifiers trained on the NSL-KDD dataset were much lower than the classifiers trained on the KDDCup99 dataset. This demonstrates how the redundant records in the KDDCup99 dataset enable algorithms to perform with higher accuracy.

Classification Metrics (Precision, Recall, and F1 scores): In order to use these metrics, the datasets' labels were reformatted to binary (0 = "normal" and 1 = "any intrusion type"). It was observed that the F1 scores (harmonic mean of the precision and recall) for the NSL-KDD trained classifiers were much lower than their KDDCup99 counterparts.

- One irregularity within my results was the relatively low classification accuracy of the "r2l" and "u2r" intrusion types by the classifiers trained on the NSL-KDD dataset. Typically, researchers have found that the NSL-KDD dataset allows for a higher classification accuracy of these intrusion types, however, my results showed the opposite. This is likely due to differences in data sampling or classifier architecture.
- The Principle Component Analysis visualization highlights one key difficulty in both datasets: there is no spatial separation of the "r2l" and "u2r" intrusion types from the rest of the data.
- Future Research:
  - Develop "stacked" ML classifiers that have a higher accuracy on the NSL-KDD dataset as it has a higher quality of data
  - Compare both datasets on a wider variety of ML algorithms

Major Citations

[1] Hettich, S. and Bay, S. D. (1999). The UCI KDD Archive [http://kdd.ics.uci.edu]. Irvine, CA: University of California, Department of Information and Computer Science.
[2] University of New Brunswick - Canadian Institute for Cybersecurity Researchers. (2015, December 9). NSL-KDD dataset. Retrieved from https://www.unb.ca/cic/datasets/nsl.html
[3] Dhanabal, L., & Shantharajah, S. P. (2015). A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, *4*(6), 446-452.
[4] Chandolikar, N. S., & Nandavadekar, V. D. (2012, September). Efficient algorithm for intrusion attack classification by analyzing KDD Cup 99. In *2012 Ninth International Conference on Wireless and Optical Communications Networks (WOCN)* (pp. 1-5). IEEE.
[5] Dertat, A. (2017, August 8). [Diagram of Artificial Neural Network]. Retrieved July 26, 2019, from https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6
[6] Saxena, S. (2018, May 11). [Precision and Recall Image]. Retrieved July 26, 2019, from https://towardsdatascience.com/precision-vs-recall-386cf9f89488
[7] Janakiev, N. (2018, October 24). Graphing ANN Keras-History Code [Python code]. Retrieved July 11, 2019, from https://realpython.com/python-keras-text-classification/#what-is-a-word-embedding

Acknowledgements

I would like to thank Dr. Ahmadi and Dr. Islam for their guidance in the completion of my project and Dr. Cobb for assisting me through ASSIP. In addition, I would like to acknowledge the UCI Knowledge Discovery in Databases Archive and the University of New Brunswick – Canadian Institute for Cybersecurity for providing me with the KDDCup99 and NSL-KDD datasets, respectively.