Predicting Process Name from Network Data

Justin Allen
Lawrence Livermore National Laboratory

Abstract: We use network data to reliably identify processes running on a connected machine. Our analysis includes a comparison of possible observables and machine-learning techniques. After modifications to eliminate training-set artifacts, our results indicate that accuracies exceeding 90% are achievable using random forests and a small number of features.

Problem:

Given a data set consisting of processes and information about the network traffic they generate, can we use the network information to predict the name of the process?

Model Selection:

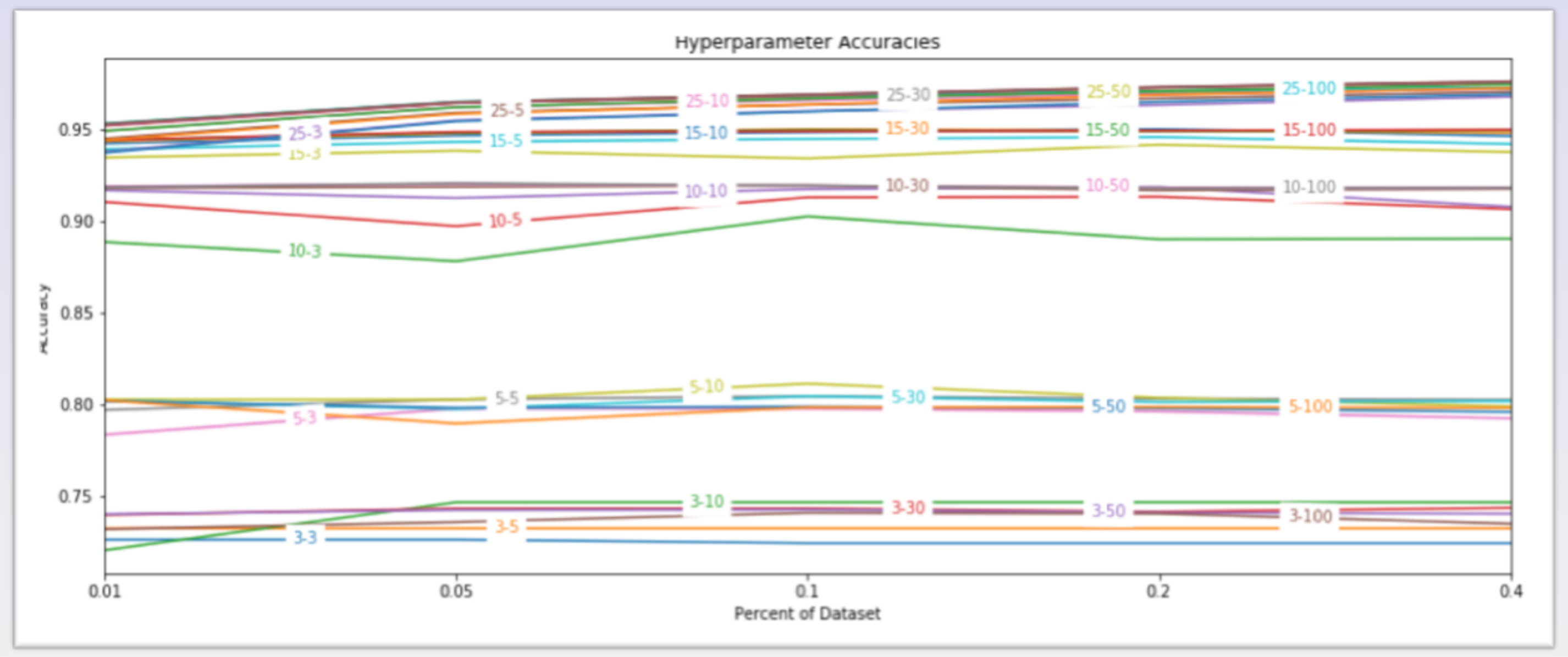
Initial tests were conducted using a random forest on a dataset consisting of ~8 million instances with 30 features. The goal was to evaluate accuracy using various hyperparameters. The hyperparameters were found using a grid-based search. Models were evaluated on a test set of one million instances.

Evaluated the effects of changing the following hyperparameters:

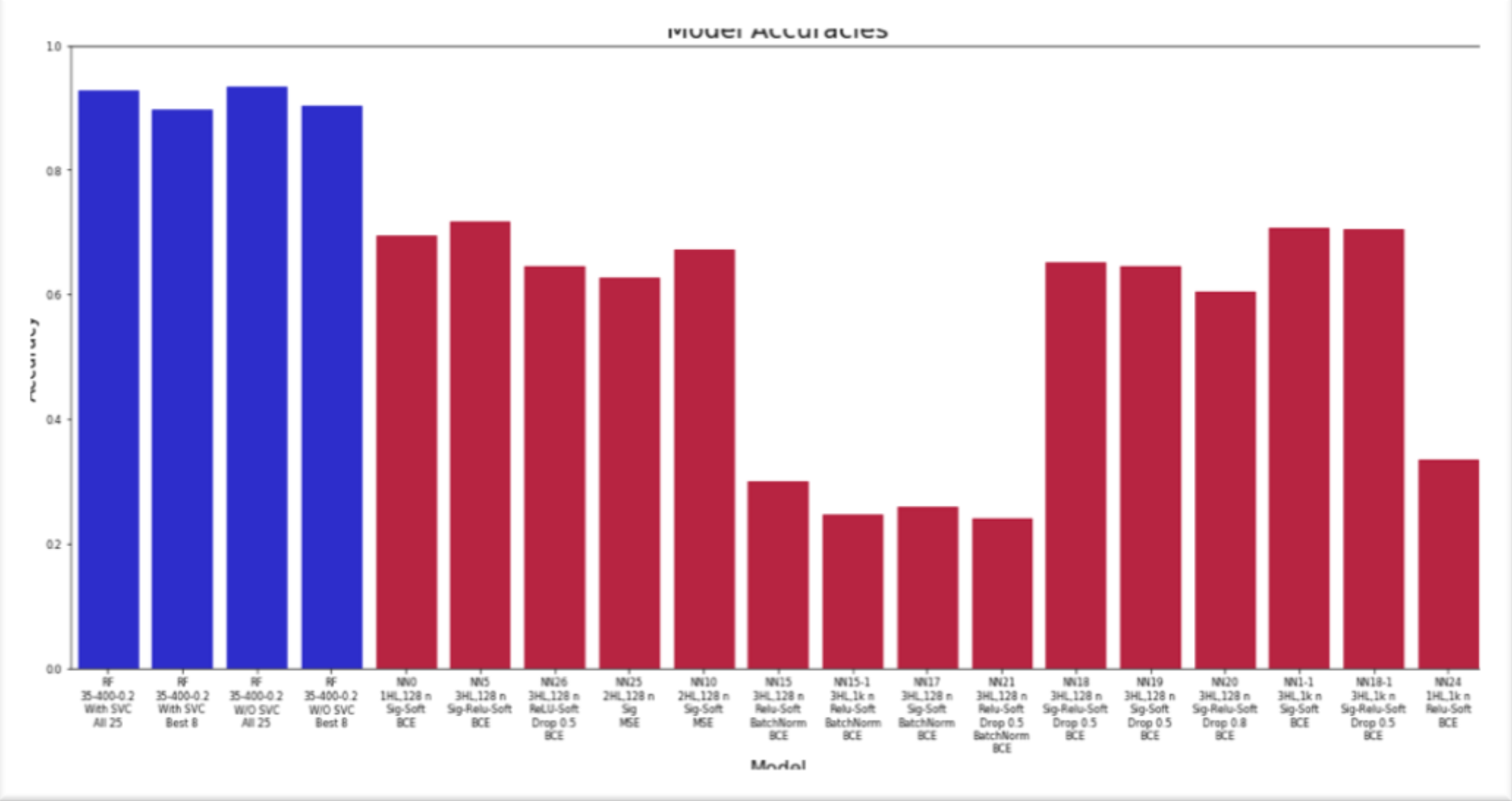
- Training set size (100,000 to 4,000,000 instances)
- Max depth (3-35 nodes)
- Number of decision tress in random forest (3-100)

Surprisingly, the number of trees and the number of data points did not noticeably affect accuracy once past a certain low threshold. The max depth, on the other hand, greatly affected accuracy, as would be expected.

Multi-Layer Perceptrons (MLP's) were also tested with differing activation functions, layers, neurons per layer, dropout percentages, error functions, and batch normalization. None of the networks tested could compete with random forests, with the highest accuracy being around 71%.



Each line represents a different hyperparameter value, with the first number being the max depth and the second number being the number of estimators. The graph shows that using any more than around 10 estimators and 5% of the dataset (around 400k samples) will not yield much better results. This greatly reduces the amount of training time needed. Max depth should be kept as high as possible.



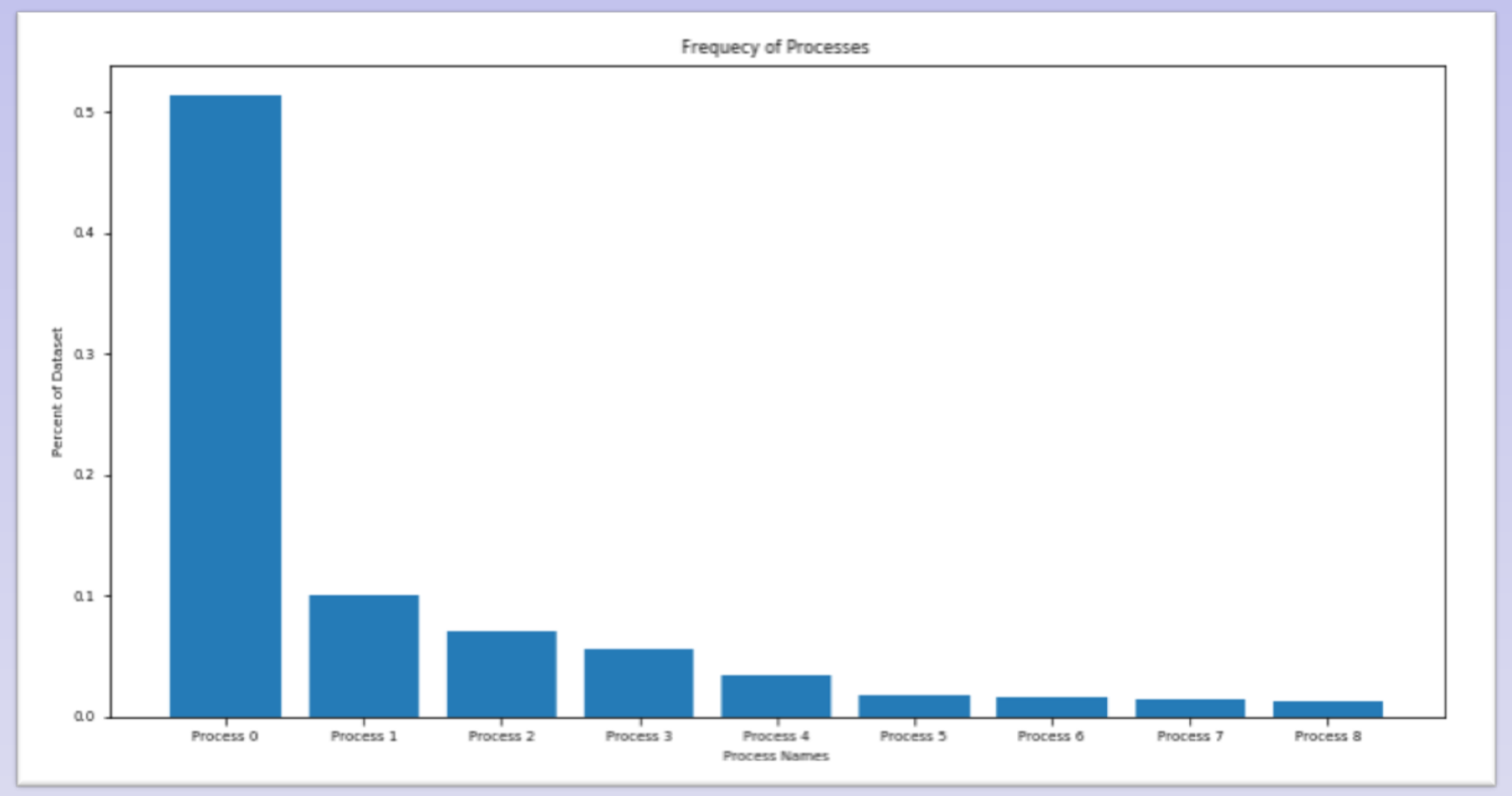
Various machine learning techniques were tested with largely varying accuracies. Random forests consistently achieved around 90% accuracy while the better MLP's averaged around 65%. The best MLP's utilized 3 hidden layers with 128 neurons each and dropout on every layer.

Adjustments and Final Results:

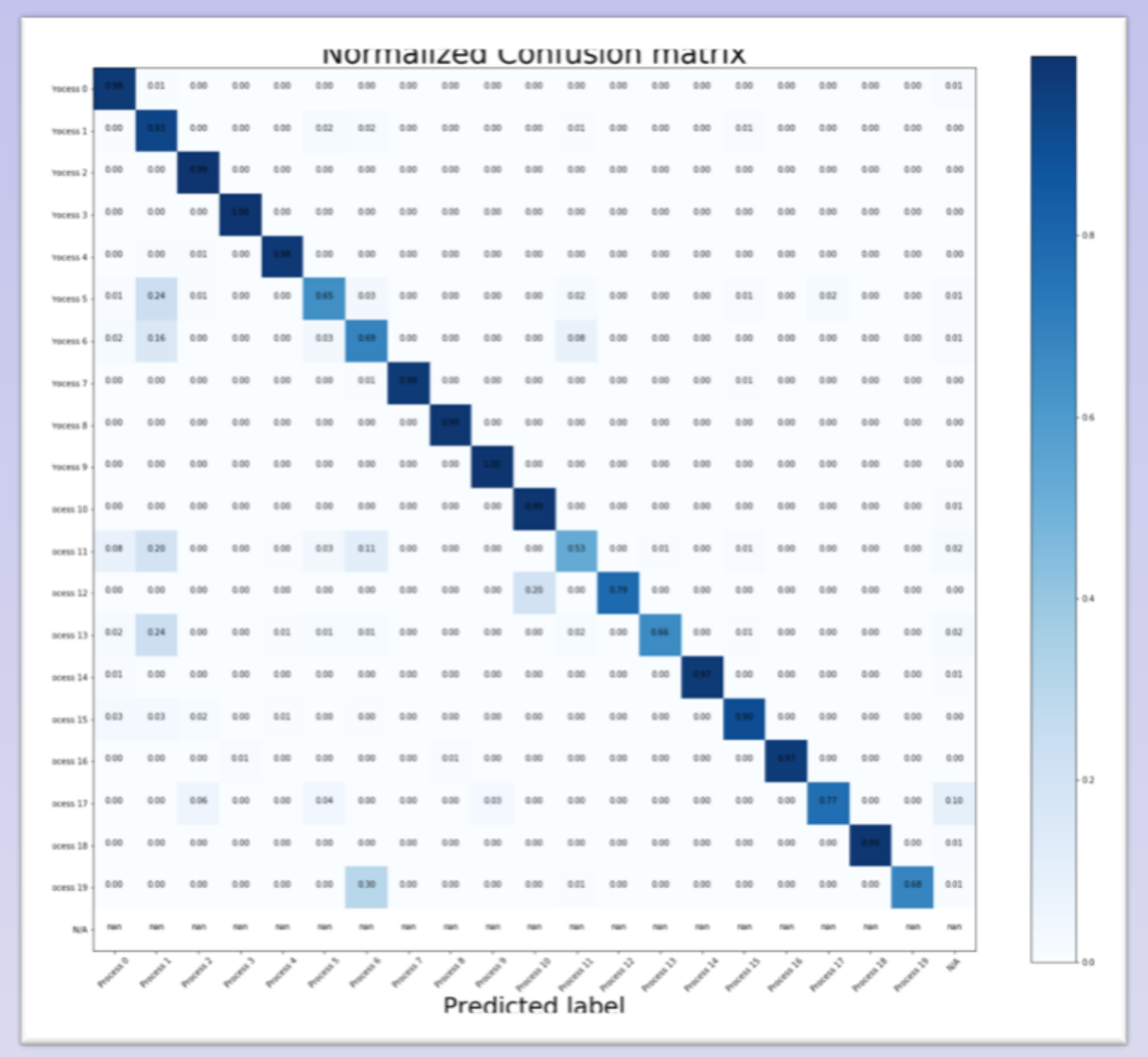
Initial experiments were able to achieve 97% accuracy in predicting process name with random forests, however, the model was relying heavily on non-generalizing features such as local IP and local port. The large effect of tree maximum depth also suggests that these models might be overfitting to the data.

Without these features, the accuracy dropped to around 93%. The forest was now only depending on the data within the packets, not IP information. Accuracy remained above 90% when trained on only the top 8 non-IP features.

Over half of the dataset was made up of Process 0. When this process was not considered, accuracy on predicting the remaining processes remained at 93%.



The percentage of the dataset that each process makes up.



The confusion matrix of the highest accuracy random forest.

Discussion:

– MLP's using large numbers of neurons (>1000 per layer) with ReLU activation performed poorly due to large activations and exploding gradients from large input data. Sigmoid activations on these large networks greatly increased performance, likely because it provided an upper bound to the activated signals. MLP's with fewer neurons (64-128 per layer) showed very similar accuracies, and the difference between ReLU and sigmoid activations disappeared.

– Using softmax activation on the last layer along with a binary cross-entropy loss function performed noticeably better than those without softmax using mean squared error. Adding dropout slightly increased performance as well.

– Batch normalization was used to reduce the size of activated signals on ReLU networks. It greatly reduced performance in all cases.

– The large effect of max depth compared to number of estimators is indicative of low variance between individual trees. In this case, it appears that the trees were overly-reliant on a single feature (likely local IP).

– The random forests still performed relatively well without IP information. For slightly lower accuracy, we were rewarded with a larger likelihood that the model would generalize well to data gathered from different IP sources.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.