Improving Anomaly-Based Intrusion Detection

Marina Moskowitz, Alina Oprea
Northeastern University, Khoury College of Computer Science

I. Problem Statement

Intrusion Detection Systems (IDSs) and Intrusion Prevention Systems (IPSs) are the most important defense tools against the sophisticated and ever-growing network attacks. As a result of unreliable test and validation datasets, there is a lot of room to improve anomaly-based intrusion detection approaches.

Existing datasets are unreliable because they are out of date and do not include recent attacks, suffer from a lack of traffic diversity and volumes, fail to cover a variety of different types of attacks, and do not take into account current trends attackers are levering to find back doors in the way technology is adapting.

As an attempt to improve anomaly-based intrusion detection approaches and performance evolutions, the publicly available dataset used for evaluation was the Intrusion Detection Evaluation Dataset (CICIDS2017), provided by the Canadian Institute of Cybersecurity [1].

II. Dataset

A. Data Collection

The CCIDS2017 dataset contains benign and the most up to date well known attacks, which resembles the true real-world data (PCAPs). It also includes the results of network traffic analysis using CICFlowMeter with labeled flows based on the time stamp, source, and destination IPs, source and destination ports, protocols, and attack.

The data collection of traffic took place over a period of five days from July 3, 2017 to July 7, 2017. Monday, July 3, 2017, includes only benign traffic. The rest of the data collected from Tuesday till Friday captures attacks including Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet and DDoS, and benign data. The data was collected in the following segments:

- Monday, Normal Activity, 11.0G
  - Benign
- Tuesday, Attacks + Normal Activity, 11G
  - Brute Force
  - FTP-Patator
  - SSH-Patator
- Wednesday, Attacks + Normal Activity, 13G
  - DoS / DDoS, DoS Slowloris
  - DoS Slowhttptest
  - DoS Hulk
  - DoS GoldenEye
  - Heartbleed Port 444
- Thursday, Attacks + Normal Activity, 7.8G
  - Morning:
    - Web Attack – Brute Force
    - Web Attack – XSS
    - Web Attack – SQL Injection
  - Afternoon:
    - Infiltration – Dropbox Download
    - Infiltration – Dropbox Download Win Vista
    - Meta Exploit Win Vista
    - Infiltration – Cool Disk – MAC
- Friday, Attacks + Normal Activity, 8.3G
  - Morning:
    - Botnet ARES
  - Afternoon:
    - Port Scan
    - NAT Process on Firewall
    - DDoS LOIT

B. Data Dimensions

The data originally had 2,830,743 rows and 79 features, including the label.

i. *Labels:*

The label consisted of 15 unique categories. In an effort to reduce the imbalance of data collected between categories, the labels were combined and encoded in 8 categories as such:

Original Labels	Categorical Label
Benign	0
FTP-Patator SSH-Patator	1
DoS Hulk DoS GoldenEye DoS slowloris DoS Slowhttptest Heartbleed	2
Web Attack Brute Force Web Attack XSS Web Attack Sql Injection	3
Infiltration	4
Bot	5
PortScan	6
DDoS	7

Figure 1: Categorical Encoding of Labels

Note: There was still are large imbalance between minority and majority classes. Efforts for resampling methods are addressed in the **Machine Learning** Section.

ii. Features

Of the 78 features, 22 were of type float64, 52 were of type int64, and 2 were of type Object.

The features of type Object (Flow Bytes/s and Flow Packet/s), were converted to type float64.

III. Pre-processing

A. Normalizing/Scaling the Data

Three different normalization and scaling techniques were evaluated against the top performing machine learning models to determine which normalization method best scaled the data. The techniques evaluated were:

- Sklearns MinMaxScaler with a feature range between 0 – 1 [2].
- Skleanrs Preprocessing StandardScaler with the mean set to 0 and standard deviation set to 1 [3].
- Imblearn.over_sample SMOTE [4].

The normalization method ultimately chosen was Sklearns MinMaxScaler function, and all of the values were scaled in a range between 0 – 1.

B. Imputing Data

Flow Bytes/s was imputed with the mean of Flow Bytes/s. No other features were dropped or imputed.

	Total	Percent
Flow Bytes/s	1358	0.047973
Label	0	0.000000
Flow IAT Min	0	0.000000
Fwd IAT Mean	0	0.000000
Fwd IAT Std	0	0.000000

Figure 2: Imputing Missing Values

C. Skewed Data

Next, outlier detection was preformed to see if any of the skewed features was of an attack label because this could be an indication for features of uncommon vulnerabilities. This is important because the labels are imbalanced, so skewed features could be a result of those features contributing to the minority label.

The following features were analyzed to determine if any of them were a strong indicator for any of the labels:

	column	skewness	unique
60	Bwd Avg Packets/Bulk	100.00	1
61	Bwd Avg Bulk Rate	100.00	1
56	Fwd Avg Bytes/Bulk	100.00	1
57	Fwd Avg Packets/Bulk	100.00	1
58	Fwd Avg Bulk Rate	100.00	1
33	Bwd URG Flags	100.00	1
31	Bwd PSH Flags	100.00	1
59	Bwd Avg Bytes/Bulk	100.00	1
49	CWE Flag Count	99.99	2
32	Fwd URG Flags	99.99	2
45	RST Flag Count	99.98	2
50	ECE Flag Count	99.98	2

Figure 3: Highly Skewed Features

There showed no indication of any feature that was highly skewed to be a strong indicator can any label. Therefore, the features in the table above whose skewness was of greater than 99% were dropped from the dataset.

D. Data Exploration

Further investigation of the following features with only two unique values, but less than 99% skewed was preformed to understand which of the features relate to which groups of attacks:

	column	skewness	unique
40	FIN Flag Count	96.46	2
41	SYN Flag Count	95.36	2
30	Fwd PSH Flags	95.36	2
59	Active Std	92.74	202826
63	Idle Std	91.90	197616
44	URG Flag Count	90.52	2
66	Label	80.30	8
61	Active Min	80.26	175670
60	Active Max	80.26	299565
58	Active Mean	80.26	326325

Figure 4: Unique Features

- When FIN Flag Count = 1, 23% of DoS attacks are related to this feature
- When SYN Flag Count = 1, 55% of Infiltration attacks are related to this feature

E. Feature Selection

A. Coefficients

The following features had coefficients of zero, meaning they did not positively or negatively relate to the Label, and therefore were removed from the dataset:

```
Subflow Fwd Packets              -0.00
Subflow Bwd Packets              -0.00
Total Backward Packets           -0.00
act_data_pkt_fwd                 -0.00
Total Length of Bwd Packets      -0.00
Subflow Bwd Bytes                -0.00
Fwd Header Length                 0.00
Fwd Header Length.1               0.00
Bwd Header Length                 0.00
min_seg_size_forward              0.00
```

Figure 5: Features with Zero Correlation

B. Collinearity

In order to determine which features were highly correlated to each other, feature selection was performed by creating a heatmap of all the features and removing one of the two features that had 99% or more correlation to each other. When determining which of the two features to remove, the feature that had the most unique values was kept. The features that had 99% or more correlation to each are shown below:

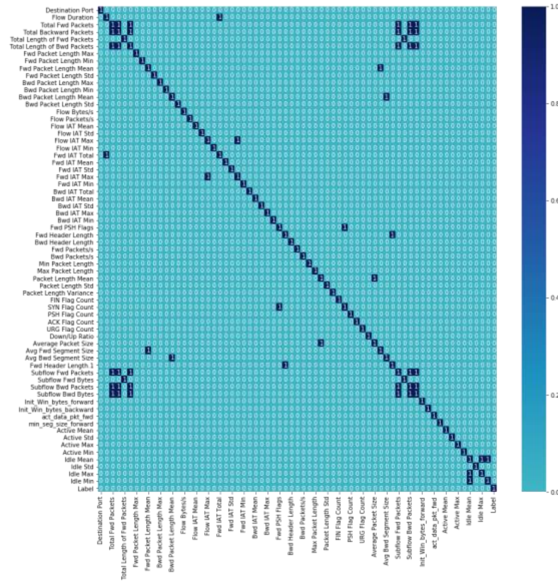


Figure 6: Highly Correlated Features

Feature 1	Feature 2	Feature Removed
Fwd IAT Total	Flow Duration	Fwd IAT Total
Subflow Fwd Bytes	Total Length of Fwd Packets	Total Length of Fwd Packets
Avg Fwd Segment Size	Fwd Packet Length Mean	Avg Fwd Segment Size
Avg Bwd Segment Size	Bwd Packet Length Mean	Avg Bwd Segment Size
SYN Flag Count	Fwd PSH Flags	Neither Removed
Average Packet Size	Packet Length Mean	Average Packet Size
Flow IAT Max	Fwd IAT Max	Fwd IAT Max
Idle Mean	Idle Max	Idle Mean

Figure 7: Highly Correlated Features Removed

Note: Neither SYN Flag Count or Fwd PHS Flags were removed. An attacker can send a segment with both flags set to see what kind of system reply is returned and thereby determine what kind of OS is on the receiving end. The attacker can then use any known system vulnerabilities for further attacks. Therefore, both of these features combined can be indication for malicious activity.

C. Random Forest Feature Importance

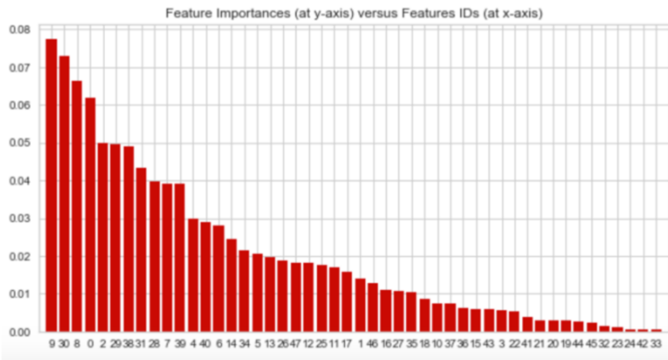


Figure 8: Visual of Random Forest Important Features

	feature	importance
33	SYN Flag Count	0.000524
42	Active Std	0.000526
24	Fwd PSH Flags	0.000594
23	Bwd IAT Min	0.001202
32	FIN Flag Count	0.001470
45	Idle Std	0.002222
44	Active Min	0.002485
19	Bwd IAT Total	0.002934
20	Bwd IAT Mean	0.002952
21	Bwd IAT Std	0.003052
41	Active Mean	0.003869
22	Bwd IAT Max	0.005255

Figure 9: Random Forest Least Important Features

Before removing the least correlated features, due diligence was preformed to determine what exactly each of these features are. After further investigation, None of these features were dropped, as the models preformed worse without them.

D. Distribution of Data

After preprocessing the data and preforming feature engineering, the data contained 28,307,43 rows and 49 features. Below is a graph of the distribution between all of the labels after preprocessing and feature extraction:

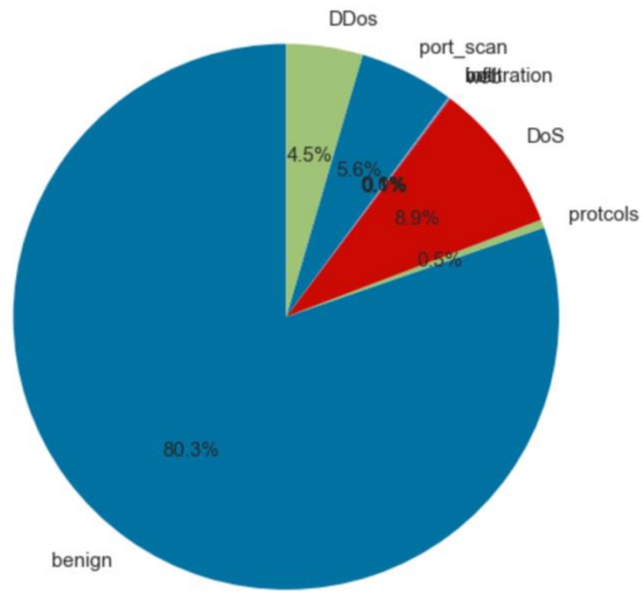


Figure 10: Label Comparison

Label	# of row of this label
Benign	2273097
Protocols	252672
DoS	252672
Web	2180
Infiltration	36
Bot	1966
Port Scan	158930
DDoS	128027

Figure 11: Label Comparison

F. Machine Learning Modeling

A. Train/Validation Split

The dataset was split into training and validation data, leaving 30% of the data as validation, in order to best determine how each of the models preform.

B. Model Selection

Various multi-class classification models were evaluated to determine which would best predict different types of attacks from network traffic data.

Due to the large class imbalance, various balancing methods were preformed selected models but results of these models did not improve.

C. Comparison of Models

Due to the large class imbalance between the labels, accuracy was not a good metric to evaluate this dataset. Various balancing methods were performed for selected models. The metrics used to determine the top preforming classifiers were precision, recall, and F1 scores. The two top preforming metrics are highlighted in the following table:

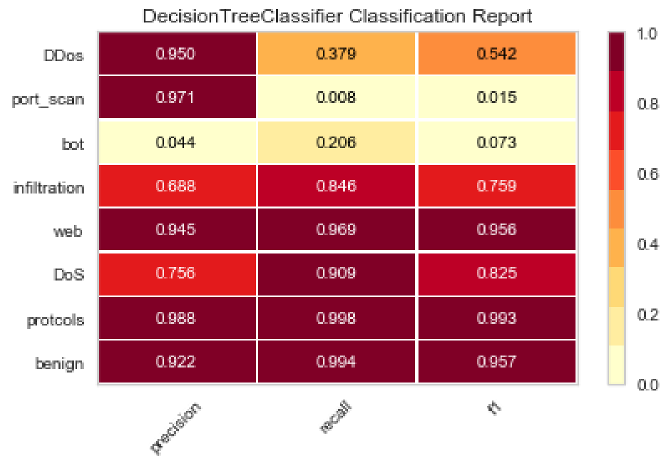
Algorithm	Precision	Recall	F1
Decision Tree	Macro Avg: .79 Weighted Avg: .97	Macro Avg: .69 Weighted Avg: .97	Macro Avg: .71 Weighted Avg: .96
Decision Tree – Class Weight Balanced	Macro Avg: .73 Weighted Avg: .87	. Macro Avg: .64 Weighted Avg: .90	Macro Avg: .61 Weighted Avg: .88
Random Forest	Macro Avg: .99 Weighted Avg: .97	. Macro Avg: .81 Weighted Avg: .97	. Macro Avg: .87 Weighted Avg: .96
Random Forest – Class Weight Balanced	Macro Avg: .94 Weighted Avg: .97	Macro Avg: .75 Weighted Avg: .97	Macro Avg: .81 Weighted Avg: .96
XGDBoost	Macro Avg: .81 Weighted Avg: .96	Macro Avg: 68 Weighted Avg: .96	Macro Avg: .70 Weighted Avg: .95
Balanced Bagging	Macro Avg: .47 Weighted Avg: .96	Macro Avg: .97 Weighted Avg: .85	Macro Avg: .51 Weighted Avg: .89
SGD – Modified Huber	Macro Avg: .44 Weighted Avg: .92	Macro Avg: .40 Weighted Avg: .92	Macro Avg: .41 Weighted Avg: .91

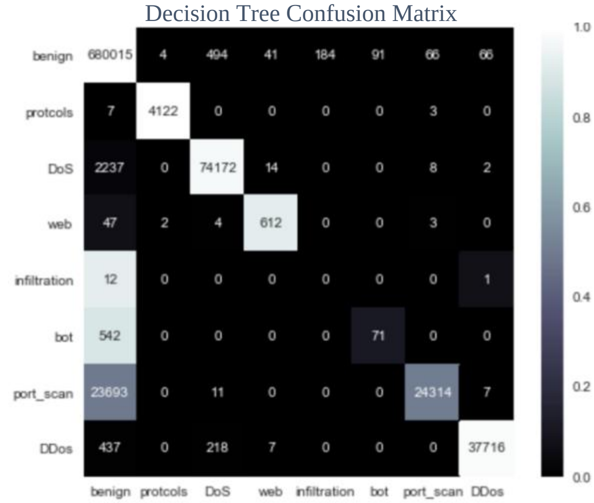
Figure 12: Model Comparison

D. Report Metrics for Top Preforming Models

1. Decision Tree

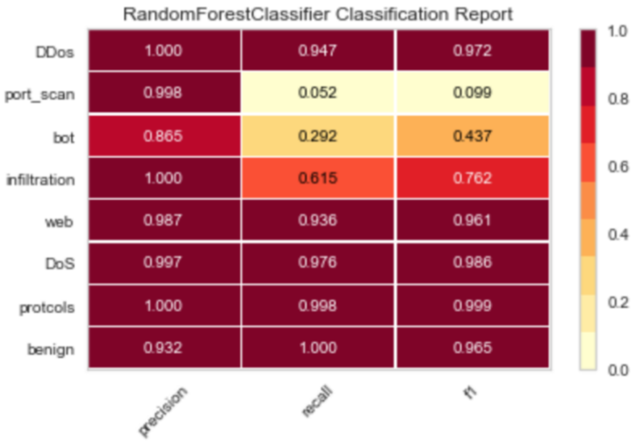
Decision Tree preformed with an 80.56% accuracy and a 0.1944 error rate.

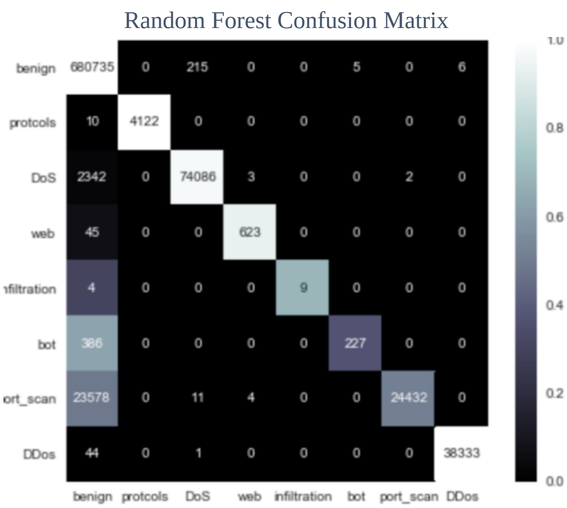




2. Random Forest

Random Forest preformed with an 80.65% accuracy and a 0.1935 error rate.





G. Conclusions

This research has successful shown that it is possible to improve anomaly-based intrusion detection approaches and performance evolutions. Pre-processing the data was first used to encode the labels, scale the data, impute any missing values, and explore any skewness in the data. Next, feature extraction was preformed to determine which features are least correlated to the label, and therefore can be removed. In the evaluation section, several different machine learning algorithms were used to train this data. Random Forest has shown excited results for an initial first step in improving anomaly-based intrusion detection. In the future, I would like to collect more data from different types of attacks, which are more up-to-date, and use machine learning to determine if I can successfully predict types of attacks within the same family. Such work would give the security world insight into how attackers are altering malware that can be indicated in network traffic and would hopefully help with predicting how network traffic will evolve in the future as new malware variants are produced.

H. References

[1] https://www.unb.ca/cic/datasets/ids-2017.html

[2] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

[3] https://scikit-learn.org/stable/modules/preprocessing.html

[4] https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.over_sampling.SMOTE.html#imblearn-over-sampling-smote