Cybersecurity Threat Classification Analysis Report

Overview

This report summarizes the analysis performed on the CICIDS2017 dataset for cybersecurity threat classification using machine learning techniques. The notebook demonstrates a comprehensive approach to data preprocessing, exploratory analysis, and model building to classify different types of network threats.

Dataset

The dataset used is from Kaggle: CICIDS2017 dataset, which contains network flow data labeled with various types of cyber threats including:

- `BENIGN(Normal Traffic) 836506`
- `DoS Hulk            115628`
- `PortScan             79433`
- `DDoS                 63938`
- `DoS GoldenEye         5065`
- `DoS slowloris         2873`
- `DoS Slowhttptest      2787`
- `Bot                    980`
- `Infiltration            16`
- `Heartbleed               7`

The dataset contains 79 features and over 1.1 million records (reduced to 50% of each column) due to Colab memory constraints) above was my final Data used.

Key Findings

Data Characteristics

- 79 features including network flow characteristics like duration, packet sizes, flags, etc.
- Highly imbalanced classes with BENIGN traffic dominating (836,506 samples)
- Some classes like Infiltration and Heartbleed have very few samples (16 and 7 respectively)

Data Preprocessing

1. **Handled Duplicates**: Removed 108,291 duplicate records
2. **Missing Values**: Dropped rows with missing values (less than 1% of data)
3. **Highly Sparse Columns**: Dropped features which contain all 0 or greater than 95 % 1 or 0.
4. **Feature Engineering**: After Feature engineering around 20 columns get dropped base on highly 0 or 1,High Corellation (99.99999),Features not adding much importance.
5. **Handled Class Imbalance**: handled Class imbalance by SMOTE
6. Train Test split was .7 for train,0.3 for test

Exploratory Analysis

- Visualized the class distribution showing severe imbalance
- Noted that most features are numerical with two categorical columns (Destination Port and Label)

Modeling Approach

Why XGBoost Was Used

XGBoost (Extreme Gradient Boosting) was selected for this classification task because:

1. **Handles Imbalanced Data Well**: With class weights parameter, it can effectively learn from imbalanced datasets
2. **High Performance**: Known for achieving state-of-the-art results on many ML tasks
3. **Feature Importance**: Provides interpretable feature importance scores
4. **Efficiency**: Handles large datasets effectively with parallel processing
5. **Regularization**: Built-in L1/L2 regularization helps prevent overfitting

Other Advantages for Cybersecurity:

- Robust to outliers and noisy data common in network traffic
- Can capture complex non-linear relationships in the data
- Handles both numerical and categorical features well
- Provides probability estimates for each class

Result

- **`Accuracy: 99.65%`**
- **`Precision: 99.81%`**
- **`Recall: 99.65%`**

- **`F1 Score: 99.71%`**

Important Findings from the Cybersecurity Threat Classification Project:

1. **High Classification Performance:**
    - The XGBoost model achieved **99.5% training accuracy** and **96.2% test accuracy**, indicating strong performance in classifying network traffic threats.
2. **Effective Feature Selection:**
    - Key features such as **Destination Port**, **Flow Duration**, **Packet Length Mean**, and **IAT Metrics** significantly influenced the model's predictive ability.
    - Dropping highly correlated features, like **Subflow Fwd Bytes** and **Subflow Bwd Bytes**, helped improve efficiency.
3. **Class Imbalance Management:**
    - I have managed class Imbalance by SMOTE.
4. **Excellent Discrimination Ability:**
    - The **accuracy of 0.99** demonstrated the model's robustness in distinguishing between different classes.
5. **Minimal False Positives/Negatives:**
    - The **confusion matrix** revealed very few misclassifications, making the model reliable for real-world cybersecurity threat detection.
6. **Scalability of XGBoost:**
    - The XGBoost classifier handled a large dataset efficiently, proving suitable for high-volume cybersecurity data.

Conclusion

The analysis presents a robust XGBoost pipeline for cybersecurity threat detection with promising initial results. However, further optimization and validation are required for production use. XGBoost is well-suited for network traffic data and cybersecurity classification tasks.