Microsoft Cybersecurity Incident Classification

Project Overview

The Microsoft Cybersecurity Incident Classification project aims to enhance the efficiency of Security Operation Centers (SOCs) by developing a machine learning model that accurately predicts the triage grade of cybersecurity incidents. The model classifies incidents into three categories: True Positive (TP), Benign Positive (BP), or False Positive (FP). This classification aids SOC analysts in prioritizing incidents and responding to threats more effectively**.**

Skills You Will Learn

Data Preprocessing and Feature Engineering Machine Learning Classification Techniques Handling Imbalanced Datasets Cybersecurity Concepts (MITRE ATT&CK Framework) Model Evaluation (Macro-F1 Score, Precision, Recall) Model Benchmarking and Optimization

Problem Statement

As a data scientist at Microsoft, your task is to build a machine learning model to classify cybersecurity incidents (TP, BP, FP) using the GUIDE dataset. This model will be used by SOCs to assist in decision-making, ultimately improving enterprise security by providing precise, context-aware recommendations.

You are expected to train the model using train.csv and evaluate its performance on test.csv with metrics like macro-F1 score, precision, and recall.

Dataset Overview

The GUIDE dataset contains records of cybersecurity incidents.

1. **Data Preprocessing Steps:** Inspect the dataset (train.csv) for feature types and distributions. Perform Exploratory Data Analysis (EDA) to identify correlations and class imbalances.
2. **Data Preprocessing:** Handle missing values using imputation or removal. Engineer new features to enhance model performance. Encode categorical features (One-Hot, Label Encoding, etc.). Normalize/standardize numerical features for better model performance.
3. **Data Splitting:** Split the data into training and validation sets (e.g., 80-20 split). Use stratification to preserve the distribution of the target variable across both sets.

4. **Model Selection and Training:** Begin with a baseline model (Logistic Regression, Decision Tree). Train advanced models like Random Forest, Gradient Boosting (XGBoost, LightGBM), and Neural Networks. Implement cross-validation to ensure consistent model performance.
5. **Model Evaluation:** Use macro-F1 score, precision, and recall to assess model performance. Tune hyperparameters using techniques like Grid Search or Random Search. Address class imbalance with methods like SMOTE or class weighting.
6. **Model Interpretation:** Analyze feature importance using SHAP values or permutation importance. Conduct error analysis to understand and minimize misclassifications.
7. **Final Evaluation on Test Set**: Evaluate the final model on the test.csv dataset and report performance metrics (macro-F1, precision, recall). Compare results with baseline performance to ensure model improvement.

Project Deliverables

Source Code:

Documented code from data preprocessing to model evaluation.

Trained Model:

The machine learning model ready for deployment.

Documentation:

A detailed report including methodology, challenges, results, and insights.

Results and Evaluation

The goal is to build a machine learning model that achieves high scores in:

Macro-F1 Score: Ensures balanced performance across all classes. Precision: Measures accuracy of positive predictions. Recall: Ensures that true positives are correctly identified.

Conclusion

Upon successful completion of the project, you will have built a model that aids SOCs in classifying cybersecurity incidents, enabling more efficient responses to security threats. You will also gain valuable experience in handling imbalanced datasets, using advanced machine learning techniques, and improving model performance with feature engineering and model evaluation strategies.