Microsoft ：

Classifying Cybersecurity Incidents with Machine Learning

Project Report

Prepared by:

Tanvi A Bamrotwar

Index:

Executive Summary:

- In this project, our goal is to improve the operational efficiency of Security Operation Centers (SOCs) by developing a machine learning model that predicts the triage grade of cybersecurity incidents using the GUIDE dataset. The model will classify incidents as true positive (TP), benign positive (BP), or false positive (FP) based on historical data and customer feedback, enabling SOC analysts to respond swiftly and accurately to potential threats.

- The project focuses on building a robust classification model to streamline incident handling and enhance security response mechanisms. By analyzing the *train.csv* dataset, we will train the model and then evaluate its performance on the *test.csv* dataset. Key metrics, including macro-F1 score, precision, and recall, will be used to measure how well the model generalizes to unseen data. The ultimate goal is to ensure that the model provides SOC analysts with precise, context-rich recommendations, optimizing cybersecurity response and improving the overall security posture for enterprise environments.

Introduction:

- **Purpose**

In today's rapidly evolving digital landscape, cybersecurity has become a critical concern for organizations worldwide. Security Operation Centers (SOCs) play a crucial role in monitoring, identifying, and responding to potential security incidents. However, SOC analysts face the challenge of managing a high volume of alerts, many of which are false positives or benign events, leading to inefficiencies and delayed responses to real threats. To address these challenges, machine learning can be leveraged to enhance SOC efficiency and optimize the triage process for incident management.

- **Objective**

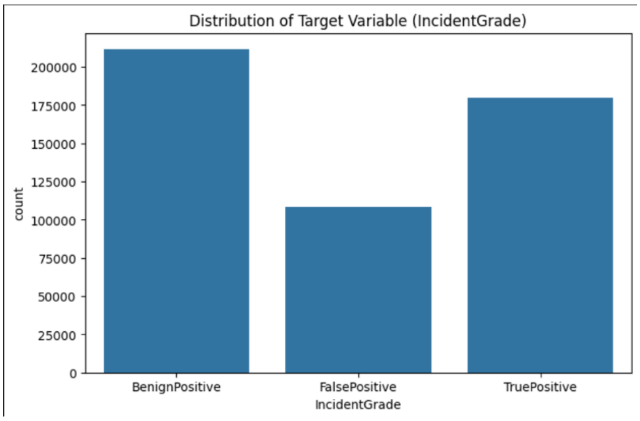
The primary objective of this project is to develop a machine learning model using the GUIDE dataset that predicts the triage grade of cybersecurity incidents. This model will categorize incidents as true positive (TP), benign positive (BP), or false positive (FP), based on historical data and customer responses. The goal is to provide SOC analysts with accurate, context-aware predictions, enabling them to focus on real threats and improve their response times. The model will be evaluated on key metrics, including macro-F1 score, precision, and recall, to ensure accuracy and generalizability.

- **Context**

By building a robust, data-driven solution, Microsoft aims to enhance the incident triage capabilities within SOCs, ultimately improving the overall security posture of enterprise environments. The project focuses on creating a predictive model that helps SOC analysts prioritize security incidents and optimize threat management. This will enable faster, more accurate responses, reduce resource strain, and improve the effectiveness of cybersecurity operations in real-world enterprise settings.

Dataset overview:-

The GUIDE dataset is large in size and it contains records of cybersecurity incidents along with their corresponding triage grades (TP, BP, FP) based on historical evidence and customer response.



Data preprocessing:-

Aim is to clean and transform the data for model training.

- **Handling Missing Data:** Identifying and addressing any missing values in the dataset .
- **Feature Engineering:** Creating new features or modifying existing ones to improve model performance, such as combining related features or encoding categorical variables.
- **Encoding:** one hot and label encoding.
- **Normalization/Standardization:** Standardization to ensure equal contribution of feature

  Outcome-Now data is cleaned and dataset is ready for model training.

Data splitting and sampling:

To split the dataset into training and validation sets for model training and evaluation

- **Train-validation split:** the data was split into training and validation sets with an 80-20 ratio , ensuring the class distribution remains consistent.
- **stratified sampling**- the technique was used to maintain the balances of classes in the training and validation sets, preventing skewed results during model training.

Model selection :-

Select and train the models to classify incidents effectively.

Baseline Models:-

Logistic Regression- Logistic regression is a statistical model used for binary classification problems. It estimates the probability that an input belongs to a certain class by using a logistic function to map predictions to a range between 0 and 1.

Decision Tree- A decision tree is a non-linear model that splits data into subsets based on feature values, forming a tree-like structure.

Advanced Models-

Random forest- An ensemble learning method that builds multiple decision trees and merges their results to improve predictive accuracy and control overfitting.

XG Boost- XG Boost is a powerful and efficient implementation of gradient boosting, which is an ensemble technique that builds sequential models, where each model attempts to correct the errors of the previous one.

Neural Networks- a deep learning model designed to capture complex patterns through multiple layers of interconnected neurons .

Model Evaluation:

Objective:-

To evaluate the models on the validation set using like micro-F1 score, precision and recall and to fine tune the models for optimal performance.

Evaluation metrics:

Macro-F1 score- assess the models performances across all classes, treating each class equally**.**

Precision an recall: Precision measures the accuracy of the models positive predictions, while recall measures its ability to identify all relevant instances.

Tuning Process

Baseline model training

the logistic regression and decision tree models are were trained as baseline models to provide a performance benchmark.

Outputs:

- **logistic regression** : achieved an accuracy 88% with the following metrics:

  Precision(macro avg):0.77 ,Recall(macro avg):0.72,F1-score(macro avg):0.74
- **Decision tree**- Achieved an accuracy 96% with the following metrics:

  Precision(macro avg):0.91 ,Recall(macro avg):0.93,F1-score(macro avg):0.91

Advanced model training:

- **Random forest model**- 5 fold cross validation; hyperparameter tubning for optimal model configuration.

  Performance-achieved 98% -accuracy, 0.96 -macro F1-score
- **XG Boost model-**hyperparameter tuning (learning rate, grid search)

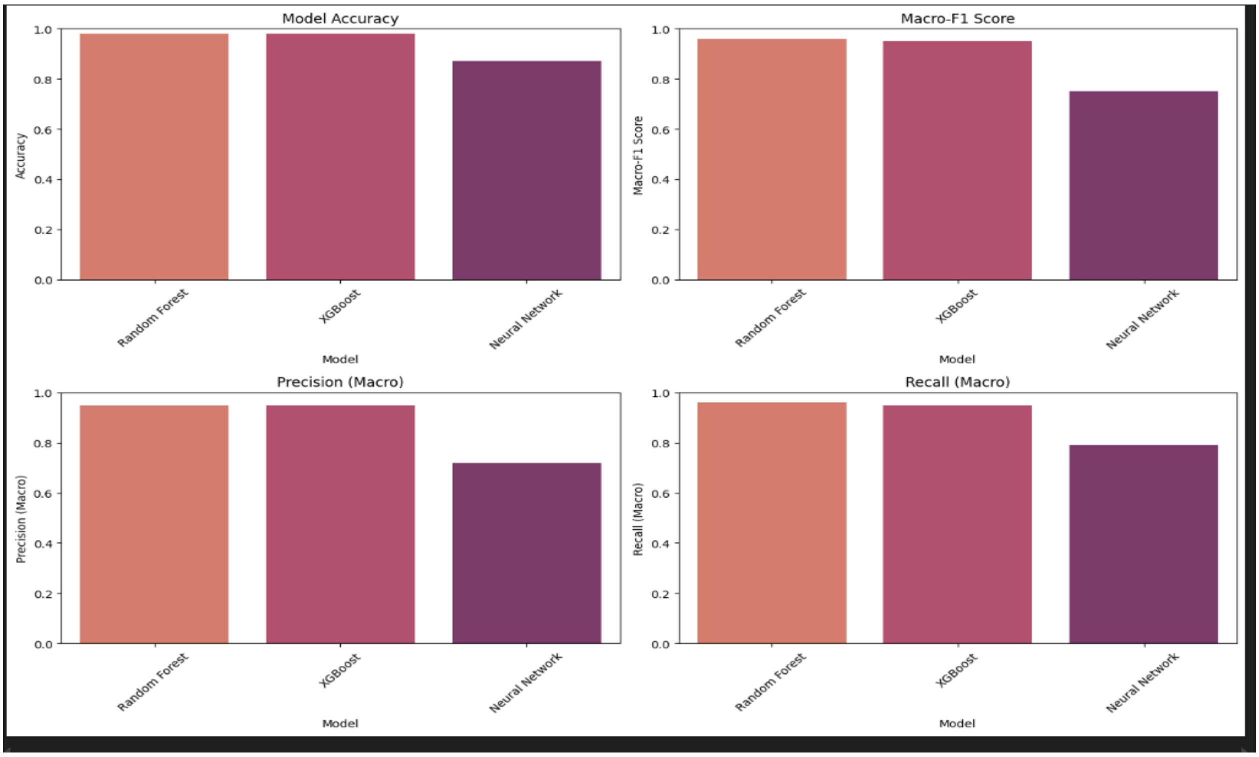
  Performance- achieved 98% accuracy ,0.95macro F1-score
- **Neural Network-**three hidden layers with dropout for regularization ,learning rate Achieved 88% accuracy , 0.77 macro F1-score.

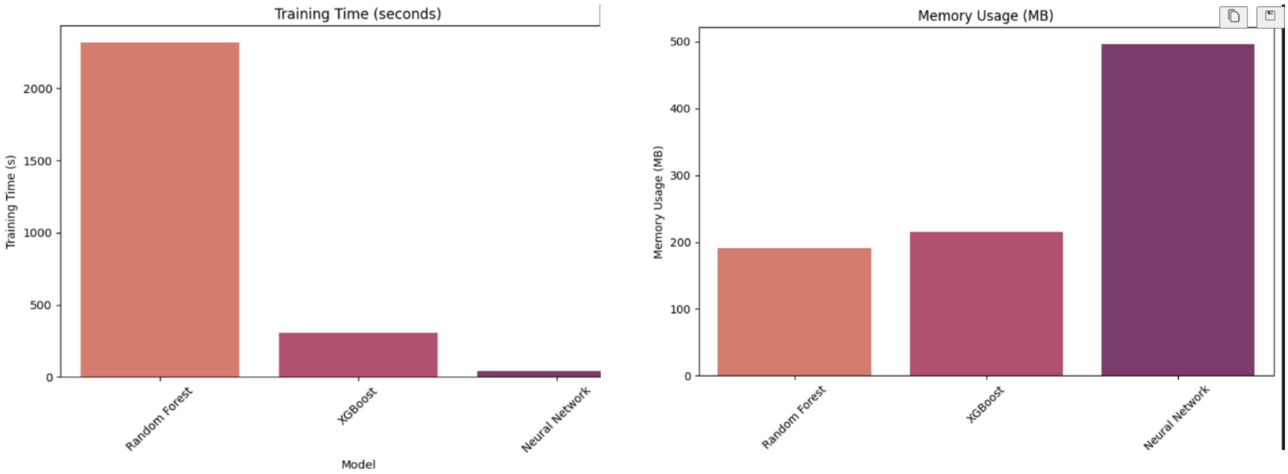
Hyperparameter Tuning: conducted using grid search for Random Forest and XG Boost, focusing on parameters like n_estimators, max_depth and learing rates.

RandomForest best Hyperparameters: {'n_estimators': 100, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 0.75, 'max_depth': 50, 'bootstrap': False}

```
Classification Report:
[[ 1607   107    31]
 [  126  1823    46]
 [  204    78 12881]]
```

The model achieved 96% accuracy ,0.92 F1-scoreand strong precision and recall across the classes.





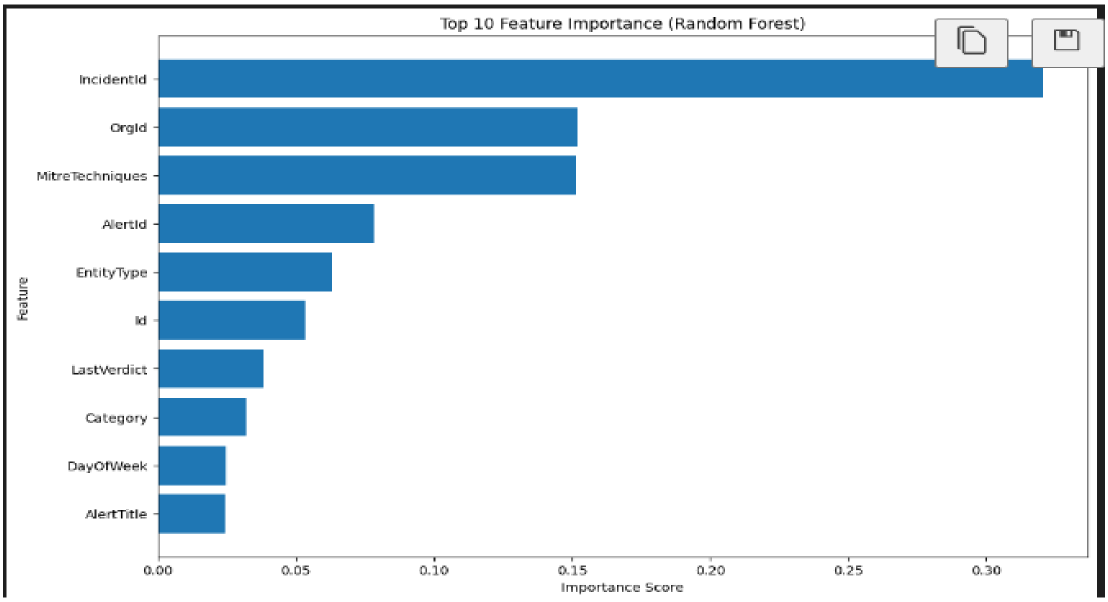
Model interpretation:

To understand the key features influencing model predictions and assess classification performance.

Techniques used:

- **Random Forest feature importance:** analyzing importance scoresderived from Random Forest model
- **Error analysis**-Reviewing misclassified cases to identify potential area for improvement.
- **Top features:**

incidentId-0.32 ,OrgId-0.15,Mitre techniques -0.15,AlterId-0.08
So certain features such as IncidentId and orgId significantly influences the models classification, particulary I distinguishing between different classes.



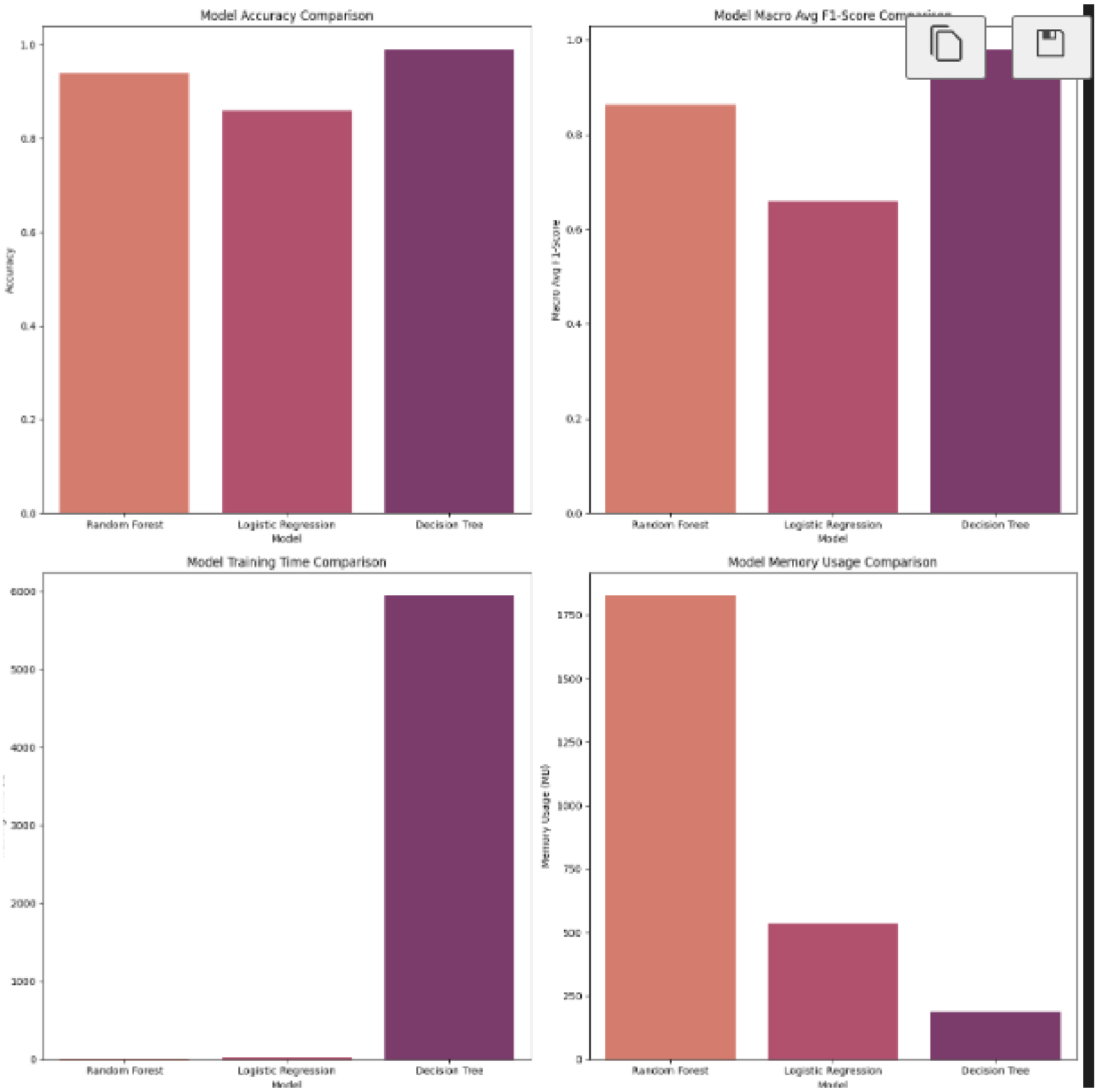
Error analysis and performance:

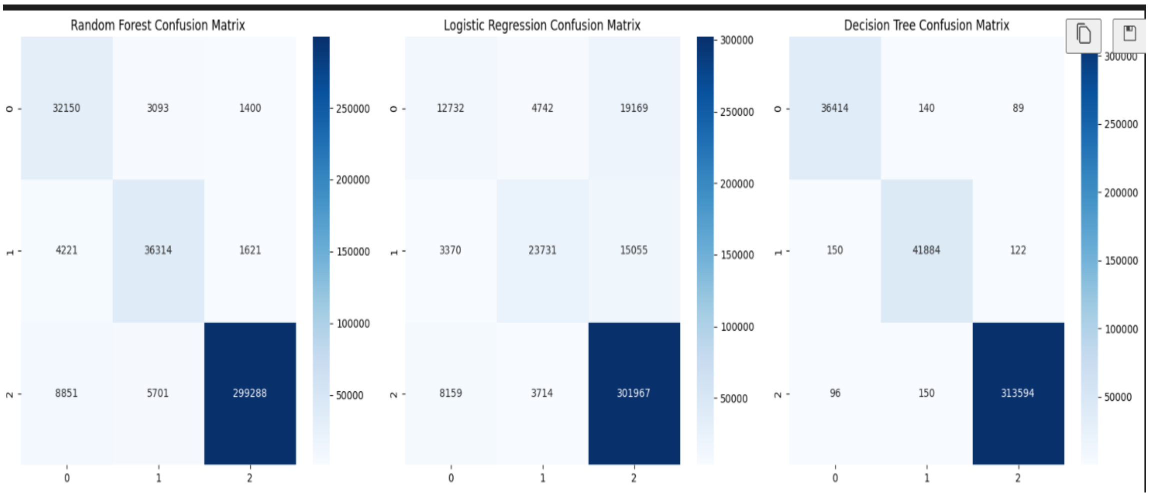
Total missclassification-592
Performance metrics: Accuracy-96%
Classification report:
Precision:- (class 0)- 0.83, (class-1)-0.91,(class-2)-0.99
Recall:- (class 0)- 0.92, (class-1)-0.91,(class-2)-0.98
F1 score:- (class 0)- 0.87, (class-1)-0.91,(class-2)-0.99
Final Evaluation on Test set: validate model performance for unseen data.
Result-
F1-score(macro):0.86
Precision(macro):0.84
Recall(macro):0.90
Accuracy:94%
Conclusion: The model shows robust performance with minimal drop in scores compare to the validation set and is suitable for real world applications

Comparison with Baseline Model: to compare the performance of advanced models with the baseline model.

1. **Logistic regression**- Accuracy of 86% , macro F1-score is 0.66
   Challenges-higher false positives , lower f1-score
2. **Decision Tree**: Accuracy of 100%macro F1-score is 0.99
   Strength-high accuracy but high training time
3. **Random Forest**: Accuracy 94%Macro F1-score is 0.86
   Advantages: significant improvement over baselines , efficient training time

Summary: Random Forest shows significant performance gains with balanced trade -off in efficiency compared to logistic Regression and Decision Tree





Challenges faced-

- **Data imbalance-**majority class (benignPositive) dominating , causing skewed predictions.
- **Model overfitting** : initial models overfitted due to class imbalance and irrelevant features.

Solution implemented-

- for data imbalance we used SMOTE technique
- Model Overfitting: cross- validation , regularization and pruning

Recommendations:

- **Integration into soc workflow:-** Deploy the Random Forest model to automate triage and enhance response times.
- **Data collection:** -Increase data collection , especially for minority classes, to improve model robustness.
- **Regular monitoring-** Continuous monitoring of model performance to adapt to evolving threats.

Conclusion: The project successfully developed a machine learning model for SOCs, achieving high accuracy and efficiency in incident triage.
next steps: Deployment , continuous improvement, and integration into border cybersecurity strategies.