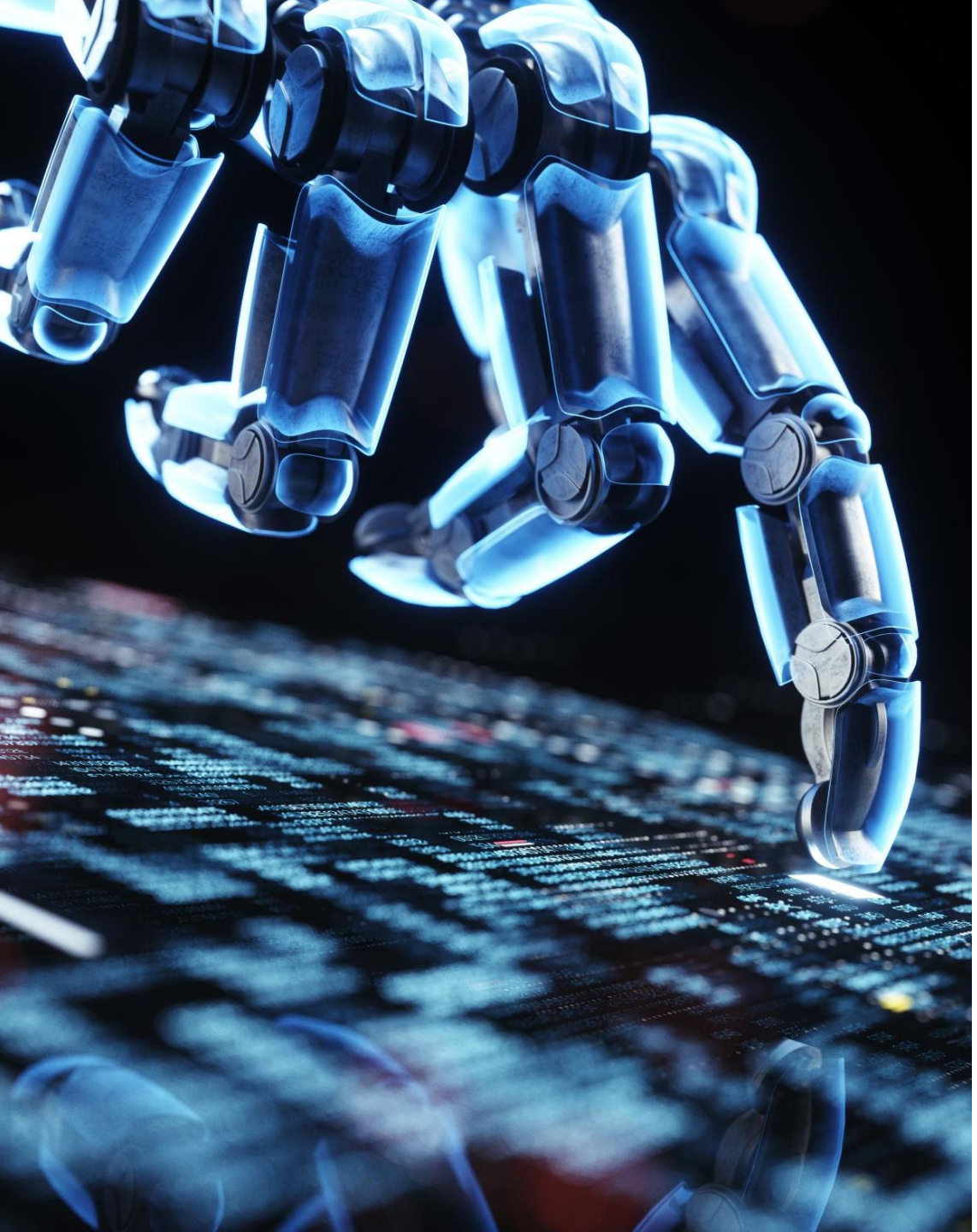
Cybersecurity Incident Classification Project

BY

SAMUELSON G



Introduction to the Domain

The cybersecurity domain focuses on safeguarding systems, networks, and data from digital threats. With the rise in cyberattacks, there is a pressing need for automated solutions to detect and classify incidents efficiently. Machine learning techniques are increasingly utilized to enhance threat detection and incident response capabilities.

Problem Statement

The objective of this project is to develop a machine learning model that accurately predicts the triage grade of cybersecurity incidents as true positive (TP), benign positive (BP), or false positive (FP) based on historical data.

Data Cleaning and Preprocessing

The notebook begins by importing essential libraries such as pandas, sklearn, seaborn, and matplotlib for data manipulation, machine learning, and visualization. Initial data checks include using **data_tr1.head()** to preview records and **data_tr1.isnull().sum()** to identify missing values. Strategies to handle missing data include dropping columns with excessive missing values and filling gaps using the mode for categorical variables.

EDA

Exploratory Data Analysis (EDA) is performed to understand the dataset's structure, including the number of rows and columns, data types, and identifying patterns or anomalies. This analysis informs subsequent preprocessing steps and feature engineering.

Feature Engineering

Feature engineering involves creating new features to enhance model performance. This may include combining related features, encoding categorical variables using techniques like one-hot encoding, and ensuring that the data is suitable for machine learning algorithms.

Statistical Significance

Statistical tests, such as the Chi-Square test, are employed to assess the independence of categorical variables. This helps identify which features are significant predictors of the target variable, ensuring that only relevant features are included in the model.

Class Imbalance Technique

To address class imbalance, techniques such as SMOTE (Synthetic Minority Over-sampling Technique) may be applied to ensure that minority classes are adequately represented, improving the model's ability to generalize.

Model Building - Base Model

The base model chosen is the Decision Tree Classifier due to its simplicity and interpretability, providing a benchmark for evaluating more complex models.

Models Used

The notebook defines several machine learning models, including:

Random Forest Classifier: An ensemble method that improves accuracy by aggregating predictions from multiple decision trees.

Decision Tree Classifier: A straightforward model that splits data based on feature values, facilitating interpretability.

XGBoost: An optimized gradient boosting algorithm effective for structured data, known for its high performance.

Model Evaluation Metric

The macro-F1 score is selected as the evaluation metric due to its ability to provide a balanced measure of precision and recall across all classes, which is crucial for imbalanced datasets.

Final Model

XGBoost is selected as the final model due to its superior performance on the validation set, demonstrating the best balance of accuracy and generalization compared to other models.

Conclusion

Feature importance analysis reveals that certain features significantly influence predictions, guiding future model improvements and providing insights for decision-making in cybersecurity incident response.

Business Suggestion/Solution

Integrating the final model into Security Operation Centers (SOCs) can automate incident triage, enhancing response efficiency and improving the overall cybersecurity posture of organizations. This solution enables quicker mitigation of threats and better resource allocation for SOC analysts.