# Project Workflow and Diagram for Anomaly Detection System

## 1. Workflow Overview

The system follows a supervised anomaly detection approach built upon a labeled dataset generated by an unsupervised Isolation Forest.  
The end-to-end pipeline is structured as follows:

1. Load Training Data (classified\_traffic\_full.csv)  
2. Extract Features and Labels (normal/anomaly)  
3. Train RandomForestClassifier  
4. Load Test Data (sample.csv)  
5. Preprocess: Clean and Normalize features  
6. Predict with the trained Random Forest model  
7. Compare predictions against true labels  
8. Generate Confusion Matrix and Evaluation Metrics  
9. Visualize Results and Export to CSV and Graph

## 2. Key Dataset Features

- Flow\_Duration: Time duration of the network flow  
- Total\_Fwd\_Packets / Total\_Backward\_Packets: Total packets in each direction  
- Fwd\_Packet\_Length\_Max / Bwd\_Packet\_Length\_Max: Max packet length per direction  
- Flow\_Bytes\_s, Flow\_Packets\_s: Byte/Packet rate of the flow  
- Fwd\_IAT\_Mean / Bwd\_IAT\_Mean: Mean inter-arrival time of packets  
- Packet\_Length\_Variance / Average\_Packet\_Size: Statistical features

## 3. Output Summary

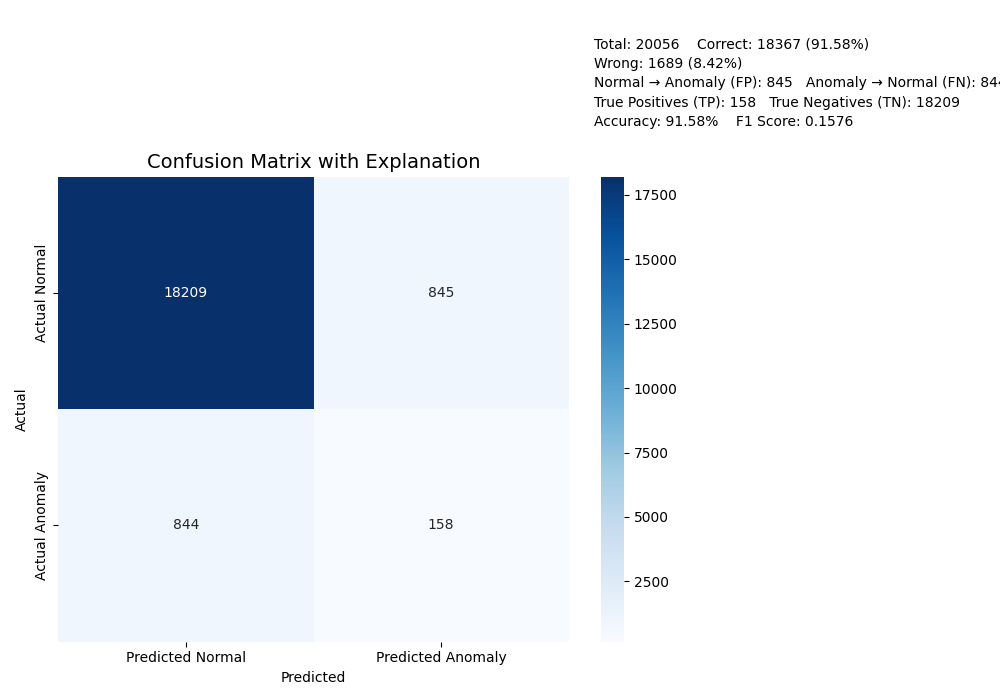
- Total Samples: 20056  
- True Positives (TP): 158 — Correct anomaly detections  
- True Negatives (TN): 18209 — Correct normal detections  
- False Positives (FP): 845 — Normal incorrectly flagged as anomaly  
- False Negatives (FN): 844 — Missed anomalies (marked normal)  
- Accuracy: 91.58%  
- F1 Score: 0.1576

## 4. Metric Equations

Accuracy = (TP + TN) / Total  
 = (158 + 18209) / 20056 ≈ 91.58%  
  
Precision = TP / (TP + FP) = 158 / (158 + 845) ≈ 0.1575  
  
Recall = TP / (TP + FN) = 158 / (158 + 844) ≈ 0.1577  
  
F1 Score = 2 \* (Precision \* Recall) / (Precision + Recall)  
 ≈ 0.1576

## 5. Confusion Matrix Diagram

The following diagram summarizes the classification results and metric values visually:



## 6. Purpose and Future Application

This anomaly detection system can play a vital role in cybersecurity, particularly for:  
- Detecting intrusions in network traffic (NIDS)  
- Preventing unauthorized access  
- Identifying malicious behavior in real-time systems  
  
In future, this system can be expanded to handle:  
- Real-time streaming data using tools like Apache Kafka  
- Advanced deep learning methods for feature extraction  
- Integration into SIEM tools for continuous monitoring