**Intrusion Detection System Analysis Using Machine Learning**

**1. Introduction**

Intrusion Detection Systems (IDS) play a crucial role in identifying and mitigating cybersecurity threats. With the rapid increase in network traffic and sophisticated cyberattacks, traditional rule-based IDS struggle to detect anomalies efficiently. Machine Learning (ML)-based IDS offer a more adaptive and intelligent approach to detecting and classifying network intrusions. This report presents an analysis of the **CICIDS dataset**, utilizing **binary and multi-class classification models** to detect and categorize network attacks.

This study evaluates the performance of multiple machine learning models for both binary classification (normal vs. attack traffic) and multi-class classification (specific attack types). The key findings highlight model effectiveness in terms of **accuracy, precision, recall, and F1-score**.

**2. Methodology**

**2.1 Dataset**

The **CICIDS dataset** consists of network traffic samples labeled as normal or various types of attacks. The dataset includes features such as flow duration, packet size, and traffic rate, which are essential for identifying network anomalies.

**2.2 Preprocessing Steps**

* Data cleaning and handling missing values
* Feature selection to remove redundant attributes
* Normalization and standardization
* Splitting data into training (80%) and testing (20%) sets

**2.3 Evaluation Metrics**

To assess the models, we utilized:

* **Accuracy** – Overall correctness of predictions
* **Precision** – Proportion of correctly predicted positive cases
* **Recall** – Ability to identify actual positive cases
* **F1-Score** – Harmonic mean of precision and recall
* **Confusion Matrices** – Breakdown of true/false positives and negatives
* **ROC-AUC Curves** – Performance comparison across thresholds

**3. Model Descriptions & Performance**

**3.1 Binary Classification (Normal vs. Attack Traffic)**

**(i) Logistic Regression**

A statistical model that predicts attack probability using a weighted sum of features.

* **Accuracy**: 99.92%
* **Precision**: 1.000
* **Recall**: 0.9984
* **F1-Score**: 0.9992

**(ii) Support Vector Machine (SVM)**

A classification algorithm that finds the optimal hyperplane for separating classes.

* **Accuracy**: 99.76%
* **Precision**: 1.000
* **Recall**: 0.9953
* **F1-Score**: 0.9976

**Finding:** Logistic Regression slightly outperforms SVM in this case.

**3.2 Multi-Class Classification (Attack Type Detection)**

**(i) K-Nearest Neighbors (KNN)**

A non-parametric algorithm that classifies samples based on their closest neighbors.

**(ii) Decision Tree**

A tree-based model that makes decisions using a series of feature-based rules.

**(iii) Random Forest**

An ensemble learning method combining multiple decision trees to improve accuracy.

**Performance Comparison (Multi-Class Classification):**

* **Random Forest** performed best due to its ensemble approach, reducing overfitting.
* **Decision Trees** showed strong classification performance but had slightly lower precision.
* **KNN** was less effective due to sensitivity to feature scaling and high dimensionality.

**4. Model Comparison & Key Insights**

| **Model** | **Type** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| --- | --- | --- | --- | --- | --- |
| Logistic Regression | Binary | **99.92%** | **1.000** | **0.9984** | **0.9992** |
| SVM | Binary | 99.76% | 1.000 | 0.9953 | 0.9976 |
| Random Forest | Multi-Class | **High** | **Best Precision** | **Best Recall** | **Best F1-Score** |
| Decision Tree | Multi-Class | Medium | Good | Moderate | Good |
| KNN | Multi-Class | Lower | Moderate | Lower | Moderate |

* **Binary classification achieved near-perfect results**, indicating that distinguishing between normal and attack traffic is highly effective.
* **Multi-class classification revealed performance variations**, with **Random Forest** emerging as the best model due to its ability to handle complex decision boundaries.
* **KNN struggled with feature scaling** and high-dimensional data, making it less suited for large-scale network traffic analysis.

**5. Conclusion & Recommendations**

This study demonstrated that **machine learning models effectively classify network intrusions**. The findings suggest:

* **Logistic Regression is an excellent choice for binary classification**, offering simplicity and high accuracy.
* **Random Forest performs best for multi-class attack detection**, benefiting from ensemble learning.
* **Feature selection and optimization** are critical for enhancing KNN performance.
* **Future Work:** Further improvements can be made using **deep learning models (LSTMs, CNNs)** or **anomaly detection techniques**.

By leveraging **ML-based intrusion detection**, organizations can enhance network security and mitigate cyber threats proactively.