Enhancing IoT Security with Machine Learning Models:

A Focus on Inference Time in Blockchain-Based Networks

December 2024

INSAT

Presented By:

Nour ElHouda Bel Haj Fraj-Nada Jarboui
Ghofrane Aloui - Shams Ben Mefteh- Arij Aguel

RT4/2

Table of Content

⊙1.Introduction

⊙2.Related Work

⊙3.Materials

⊙4.Experimental results

⊙5.Discussion

⊙6.Conclusion and future scope

Introduction

Introduction

- Network security in IoT environments faces growing challenges due to rising cyber threats and the need for rapid decision-making. Integrating blockchain enhances security through decentralization and immutability, while machine learning (ML) improves intrusion detection and system performance.

- This study evaluates the performance of various ML models in blockchain-enabled IoT networks, focusing on prediction accuracy and inference time,two key factors for real-time systems.

Related Work

Related Work

Traditional Approaches:

- Signature-based detection (e.g., antivirus tools).
- Limitations: Inability to detect unknown attacks.

Modern Approaches:

- **Blockchain-Based Security**

provides decentralized, tamper-proof data storage, ensuring secure communication and data integrity

- **Machine Learning Integration**

enhance threat detection by analyzing patterns and adapting to new threats

Related Work

Existing Gaps:

- High latency in ML inference for real-time IoT.
- Lack of integration studies between ML and blockchain.
- Need for efficient algorithms to process and analyze massive data volumes

Discussion

The convergence of blockchain and machine learning offers promising solutions to enhance IoT network security.

This research evaluates ML models, focusing on their prediction accuracy and inference time, to address the challenges of real-time IoT applications.

The goal is to identify efficient and adaptive models for robust IoT network security.

Materials

Blockchain:

How role does it play in this study ?

- Security
- Transparency
- Tamper-Proof

Materials

KDD Cup-99

a standard benchmark for network intrusion detection
Despite its relevance, the dataset has been criticized for its high redundancy and imbalanced data, which can lead to biased evaluation of ML models.

NSL-KDD:

Enhanced KDD CUP 99 dataset with reduced redundancy.
Ensures a more balanced ditribution of attack and normal records, making it suitable for training and evaluating intrusion detection systems.

Why those two datasets

These two datasets offer:

- Diverse attack scenarios.
- Comprehensive feature representations.
- Standardized benchmarking capabilities.
- Extensive coverage of network traffic patterns.

Both datasets enable rigorous testing of ML models, particularly for evaluating accuracy and inference time.

Experimental Analysis

Analysis Metrics:

This work centers on the evaluation of several machine learning (ML) models in blockchain-enabled environments, specifically targeting their effectiveness in ensuring the security of real-time IoT networks. The analysis focuses on two critical metrics:

1. Accuracy: the proportion of the total number of predictions that were correct. It is given by the relation:

$$\text{Accuracy}=\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$$

2.Inference time: refers to the time it takes for a machine learning model to process input data and generate a prediction or decision.

$$Inference\ Time = \frac{T_{prediction\ Start} - T_{prediction\ End}}{Total\ Samples}$$

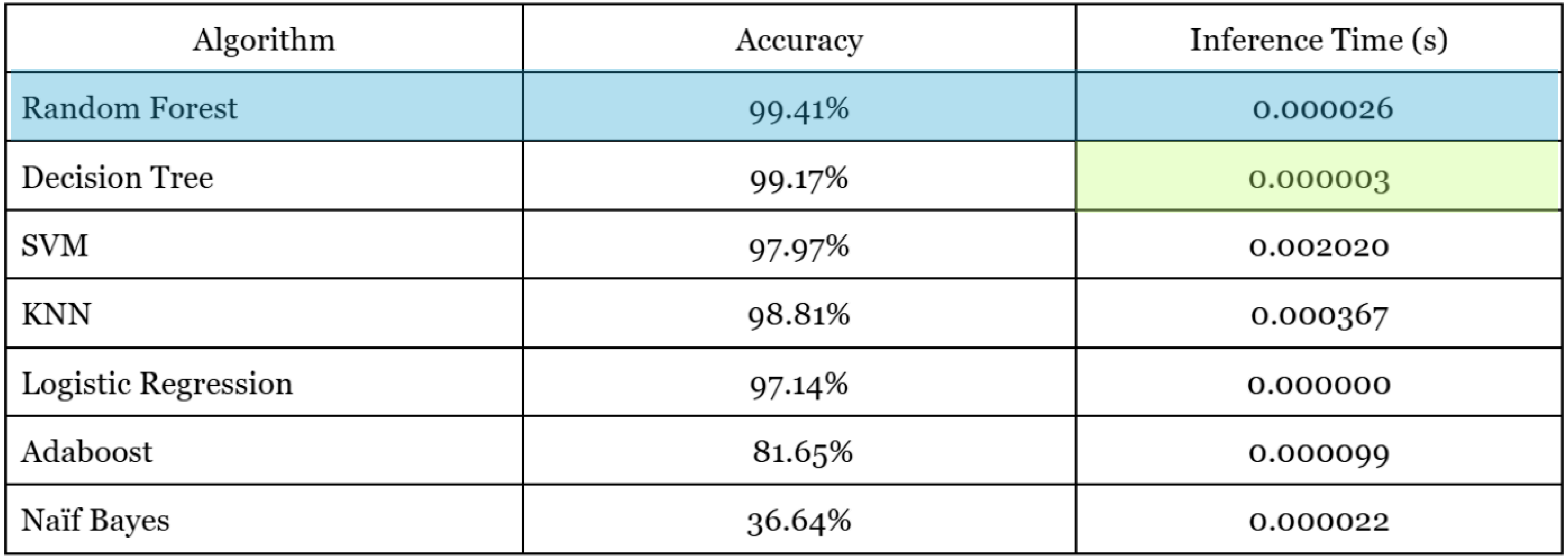
This research was carried out using the Python programming language. Several widely used Python libraries (Scikit-learn, Pandas, Numpy, Sklearn, Matplotlib, Seaborn, TensorFlow, and Keras) In order to detect intrusions in IoT networks, we developed machine learning models using these libraries.

To implement the proposed Models, these steps were followed

- Loading and Preprocessing the Datasets: which includes removing redundancies, converting categorical data into a numerical form, and normalizing the numerical data.
- Splitting the data into training (85%) and testing (15%) sets.
- Training then evaluating the models.

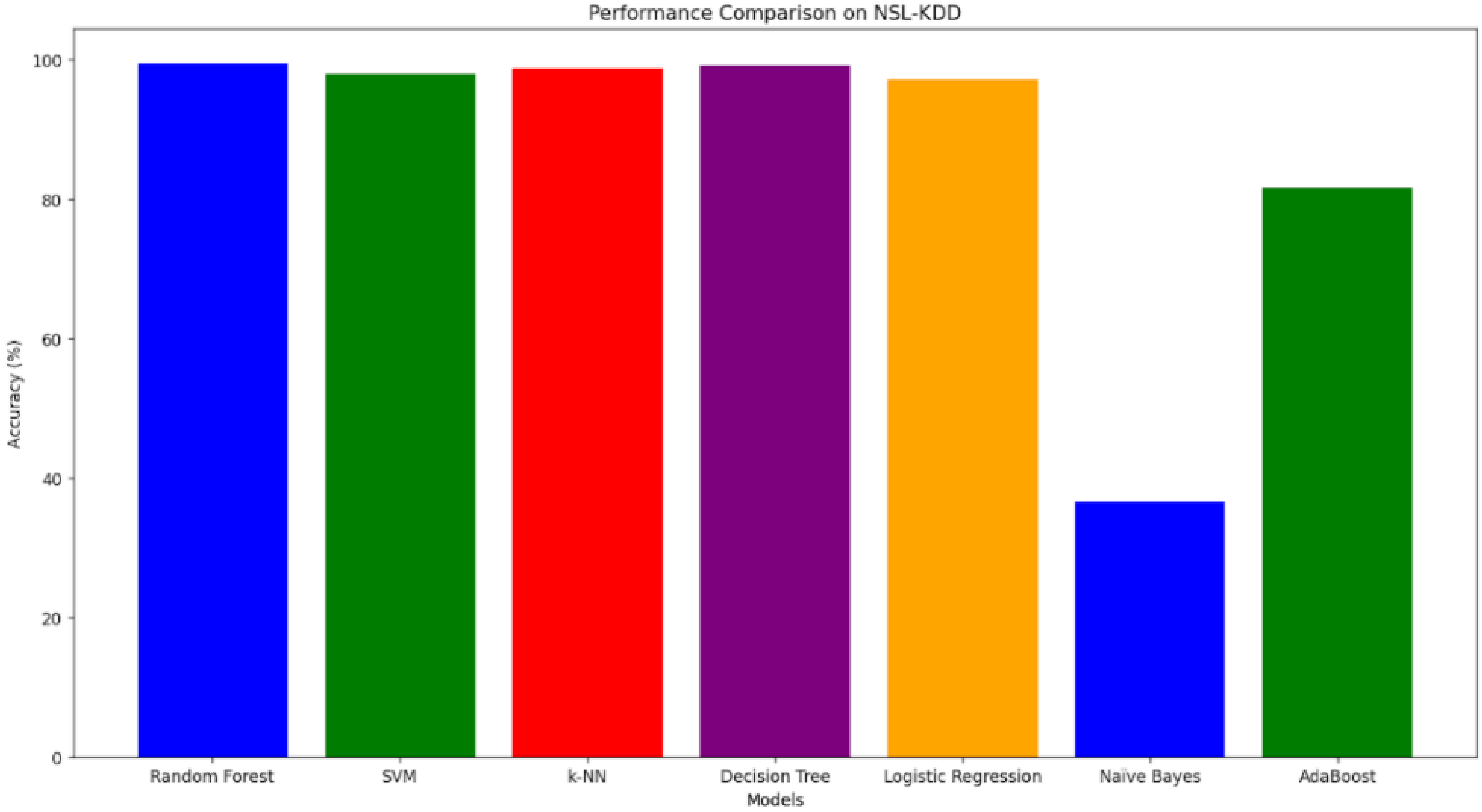
Experimental Results

NSL-KDD (KDDmerged (KDDTrain+(125 973 samples) et KDDTest+(22 544 samples)=148 517 samples) with o.15 test size)



Algorithm	Accuracy	Inference Time (s)
Random Forest	99.41%	0.000026
Decision Tree	99.17%	0.000003
SVM	97.97%	0.002020
KNN	98.81%	0.000367
Logistic Regression	97.14%	0.000000
Adaboost	81.65%	0.000099
Naïf Bayes	36.64%	0.000022

NSL KDD


Performance Comparison on NSL-KDD
Accuracy (%)
0
20
40
60
80
100
Random Forest
SVM
k-NN
Decision Tree
Logistic Regression
Naïve Bayes
AdaBoost
Models

Overall Performance of Algorithms

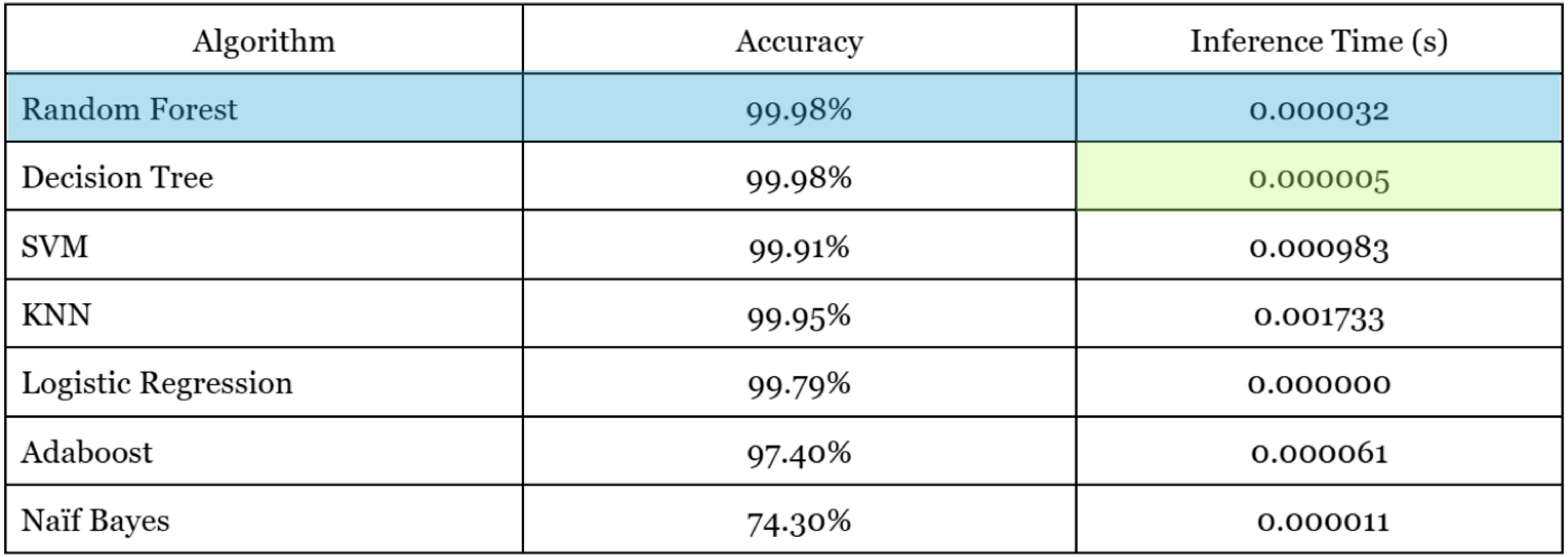
- Random Forest achieves the highest accuracy (99.41%),followed closely by Decision Tree (99.17%) and KNN (98.81%).
- Adaboost (81.65%) and Naïve Bayes (36.64%) show relatively poor performance, making them unsuitable for this task.

Inference Time

- Decision Tree has the shortest inference time (0.000003 s), making it ideal for real-time systems.
- Random Forest, while slightly slower (0.000026 s), remains competitive due to its exceptional accuracy, making it suitable for high-reliability systems.
- SVM (0.002020 s) and KNN (0.000367 s) have relatively higher inference times, limiting their applicability in time-critical environments.

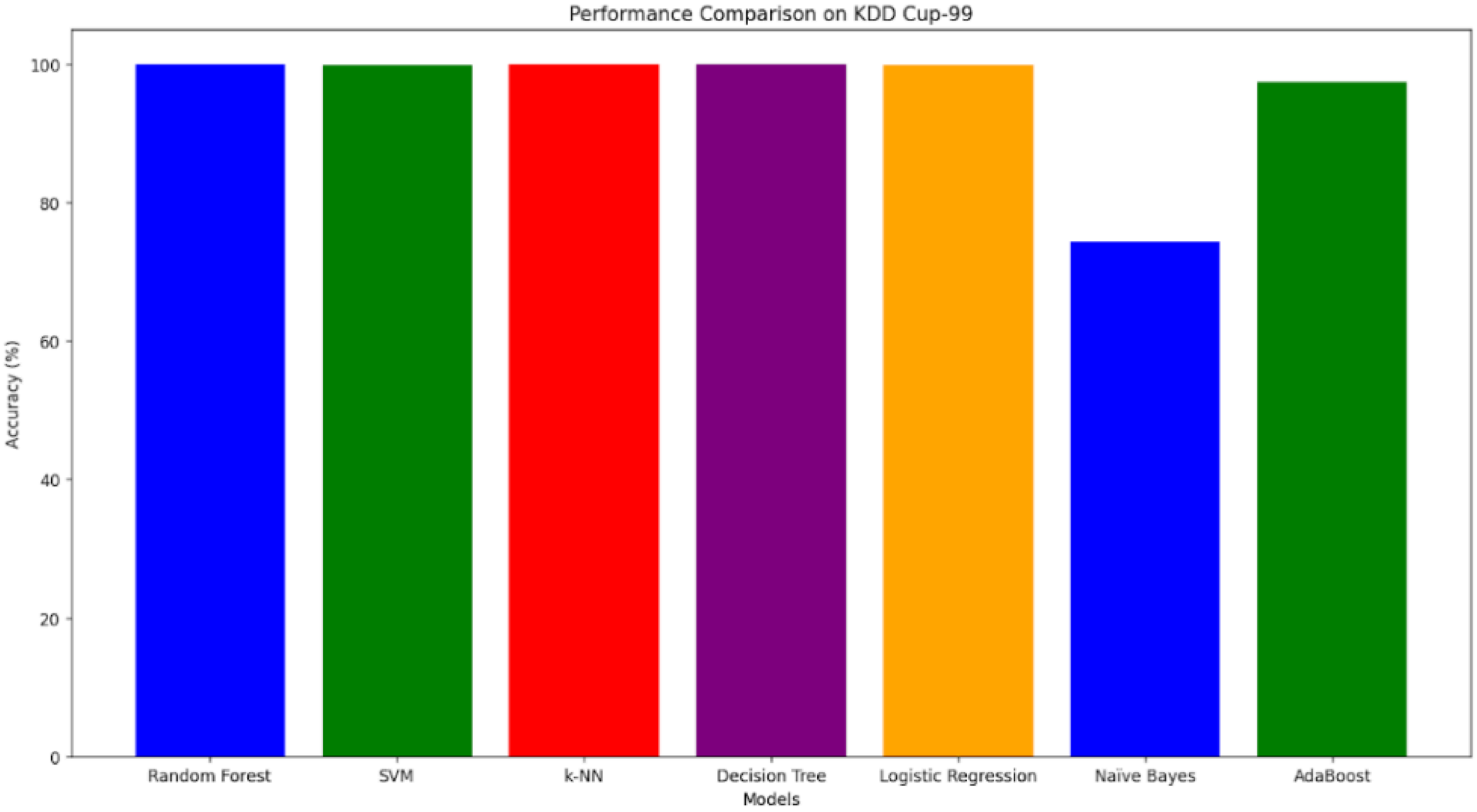
Experimental Results

KDD CUP 99: 1 074 992 samples with 0.15 test size



Algorithm	Accuracy	Inference Time (s)
Random Forest	99.98%	0.000032
Decision Tree	99.98%	0.000005
SVM	99.91%	0.000983
KNN	99.95%	0.001733
Logistic Regression	99.79%	0.000000
Adaboost	97.40%	0.000061
Naïf Bayes	74.30%	0.000011

KDD Cup 99


Performance Comparison on KDD Cup-99
Accuracy (%)
0
20
40
60
80
100
Random Forest
SVM
k-NN
Decision Tree
Logistic Regression
Naïve Bayes
AdaBoost
Models

Overall Performance of Algorithms

- Random Forest and Decision Tree achieve the highest accuracy (99.98%), followed by KNN (99.95%) and SVM (99.91%).
- Logistic Regression (99.79%) and Adaboost (97.40%) show slightly lower performance, while Naïve Bayes has the weakest performance (74.30%).

Inference Time

- Decision Tree has the fastest inference time (0.000005 s), making it ideal for systems requiring rapid execution.
- Random Forest follows with a slightly higher time (0.000032 s), but remains competitive due to its high accuracy.
- Naïve Bayes, despite having a fast inference time (0.000011 s), is ineffective due to its low accuracy.

Discussion

- The adoption of hybrid techniques that integrate blockchain-based data management and access control systems with machine learning algorithms can provide a robust and effective solution to IoT security challenges.

- The results of this study demonstrate how machine learning Algorithms can improve IoT system security. However, selecting an appropriate algorithm relies on a number of variables, including the volume and kind of data, processing speed, and the IoT system's resource limitations. Therefore, it is essential to assess various machine learning Models and Choose the one that best meets the needs of the IoT system.

The findings of this study demonstrate that **Random Forest** and **Decision Tree** can significantly enhance intrusion detection systems in blockchain and IoT networks.

- For real-time IoT networks with limited resources, Decision Tree is the optimal choice.
- For systems requiring high reliability and greater resources, Random Forest excels with its superior accuracy.

Conclusion

- By leveraging blockchain's decentralized nature and ML's predictive capabilities, we developed a system capable of detecting anomalies and malicious behavior efficiently.
- The accuracy of machine learning models plays a pivotal role in ensuring that potential threats are detected, thereby maintaining the reliability of IoT devices.
- Similarly, inference time is equally critical in such real-time environments, where rapid detection and response are necessary to prevent breaches and disruptions in the network.

- Our results demonstrate that by optimizing both accuracy and inference time, the proposed solution can serve as a robust framework for securing IoT networks: In both the NSL-KDD and KDD CUP 99 datasets, Random Forest and Decision Tree perform the best. Random Forest offers high accuracy, making it ideal for applications requiring reliability, while Decision Tree is faster, making it suitable for real-time systems.
- Blockchain further complements this system by ensuring tamper-proof logging, traceability, and trust, which are essential in distributed IoT environments.

Future Scope

Thank you