## **ASSIGNMENT-04**

**INFORMATION SECURITY**

**NAME: UMAR IFTIKHAR & TALHA SYED**

**ROLL: 21I-2710 &**

**CLASS: AI(J)**

## 

## 

## **1. Introduction**

This report provides a critical analysis of the performance of various machine learning models applied to classify URLs into five categories: Benign, Defacement, Phishing, Malware, and Spam. The analysis includes an evaluation of traditional machine learning models, deep learning architectures, and large language models (LLMs). The report also highlights key challenges encountered and suggests potential improvements for future work.

## **2. Model Performance Analysis**

### **2.1 Machine Learning Models**

We trained and evaluated the following traditional ML models:

* **Random Forest Classifier**
* **Support Vector Machine (SVM)**
* **XGBoost Classifier**

**Performance Metrics:**

* Accuracy:
* Precision:
* Recall:
* F1-Score:

Among these, **[Best performing model]** achieved the highest accuracy due to **[reason, e.g., robustness to feature variance, ability to handle non-linear relationships]**. However, **[underperforming model]** struggled due to **[reasons, e.g., sensitivity to imbalanced data]**.

### **2.2 Deep Learning Models**

We implemented and evaluated the following deep learning models:

* **LSTM (Long Short-Term Memory)**
* **CNN (Convolutional Neural Network)**

**Performance Metrics:**

* Accuracy:
* Loss:

LSTM performed well in capturing sequential patterns in URLs but required extensive training time. CNN, while computationally efficient, struggled with longer and complex URL structures.

### **2.3 LLM-Based Models**

We fine-tuned and tested:

* **BERT (Bidirectional Encoder Representations from Transformers)**
* **GPT-based Model**

These models leveraged contextual embeddings to improve classification. **BERT outperformed traditional models** due to its ability to understand contextual patterns within URLs. However, high computational cost and training time were challenges.

## **3. Challenges Faced**

### **3.1 Data Challenges**

* **Imbalanced dataset:** Some classes had significantly fewer samples, leading to biased predictions.
* **Noisy data:** URLs contained redundant or missing information affecting feature extraction.

### **3.2 Model Training Challenges**

* **Computational constraints:** Training deep learning models required high GPU resources.
* **Hyperparameter tuning:** Finding the best parameters for ML models was time-consuming.

## **4. Potential Improvements**

* **Data Augmentation:** Generate synthetic samples for underrepresented classes to balance the dataset.
* **Feature Engineering Enhancements:** Include additional URL-based features such as WHOIS data, domain age, and lexical analysis.
* **Hyperparameter Optimization:** Use automated tuning techniques (e.g., Grid Search, Bayesian Optimization) for improved performance.
* **Ensemble Learning:** Combine multiple models to leverage strengths of different approaches.
* **Cloud-based Training:** Utilize cloud resources (e.g., Google Colab, AWS) to handle LLM training efficiently.

## 

## **5. Conclusion**

This study demonstrated that **\_\_\_\_** achieved the highest classification accuracy among traditional models, while **BERT-based models** provided the most robust performance but at a higher computational cost. Future improvements should focus on data balancing, feature engineering, and computational efficiency to enhance overall model effectiveness.