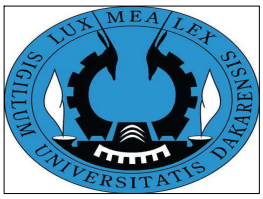
Université Cheikh Anta Diop



Ecole Supérieure Polytechnique

Département Génie Informatique

Année Universitaire 2023-2024

Ibrahima Ndendé Sow **M1GLSI**

Moustapha Mangane

Rapport sur le Projet de Détection de Malware basée sur le Machine Learning

Introduction

Le projet de Détection de Malware basé sur le Machine Learning vise à développer un système capable de classifier les fichiers exécutables en tant que malveillants ou non malveillants en utilisant divers algorithmes de machine learning. Le pipeline du projet comprend l'importation du dataset, le prétraitement des données, la division des données, le choix des algorithmes, l'entraînement des modèles, l'évaluation des performances, la sélection du meilleur modèle, et enfin, le déploiement du modèle à l'aide de Flask.

Importation du Dataset

Le dataset utilisé contient des informations extraites à partir de 137 444 exécutables, avec huit caractéristiques pour chaque exécutable. Ces caractéristiques incluent des informations telles que l'adresse de départ, la version du linker, la version de l'image, etc. Chaque fichier est étiqueté comme malveillant (1) ou non malveillant (0).

Prétraitement des Données

Les données sont prétraitées en séparant le dataset en features (caractéristiques) et labels (étiquettes). Les features sont les différentes caractéristiques extraites des fichiers exécutables, tandis que les labels indiquent si le fichier est malveillant ou non.

Division des Données

Les données sont divisées en ensembles distincts pour l'entraînement (70%) et les tests (30%) à l'aide de la fonction train_test_split de scikit-learn.

Choix des Algorithmes

Quatre algorithmes de machine learning pertinents ont été sélectionnés pour résoudre le problème de classification de malware. Les algorithmes choisis sont **Random Forest** (rf), **Support Vector Machines** (svm), **Gradient Boosting** (gb), et **Neural Network** (nn).

Entraînement des Modèles

Chaque modèle est entraîné sur l'ensemble d'entraînement à l'aide des données prétraitées.

Évaluation des Modèles

Les performances des modèles sont évaluées en utilisant des métriques telles que l'accuracy, la précision, le rappel et le F1-score sur l'ensemble de test.

Sélection du Meilleur Modèle

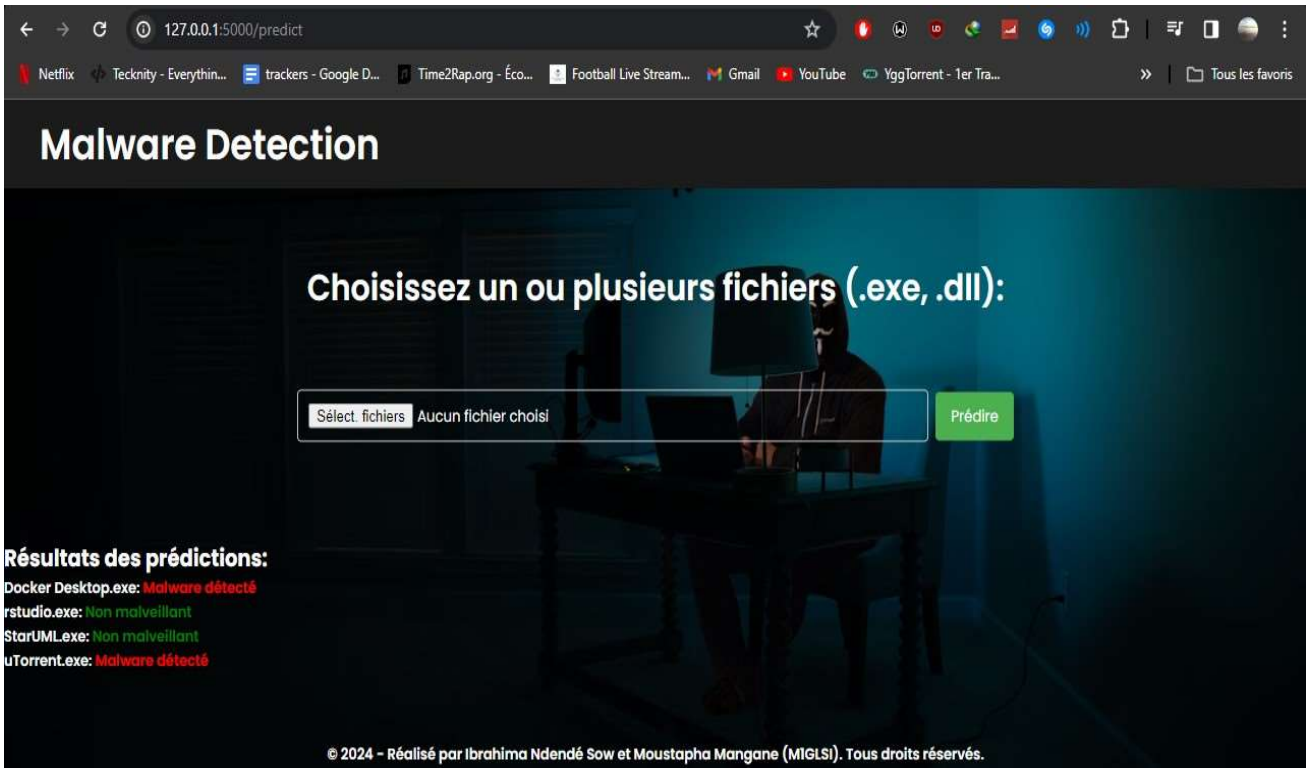
Le modèle présentant la meilleure précision est sélectionné et enregistré pour une utilisation future. Dans ce cas, le modèle **Random Forest** a été identifié comme le meilleur modèle.

Déploiement du Modèle avec Flask

Le modèle sélectionné a été déployé à l'aide du framework Flask. L'application web permet aux utilisateurs de soumettre des fichiers exécutables (.exe, .dll) pour une analyse en temps réel par le modèle de détection de malware. L'interface utilisateur conviviale facilite l'interaction avec le modèle déployé.

Résultats du Test

Les résultats du test ont été obtenus en soumettant des fichiers exécutables à l'application web déployée. Les prédictions du modèle, indiquant si un fichier est malveillant ou non malveillant, sont affichées à l'utilisateur. Un exemple de capture d'écran des résultats du test avec 4 fichiers uploadés est présenté ci-dessous.



Problèmes Rencontrés et Solutions

- **Taille du Dataset**: Le dataset initial pourrait être limité en taille, affectant la capacité des modèles à généraliser. Une solution pourrait être l'augmentation du dataset ou l'utilisation de techniques de régularisation.

- **Choix des Caractéristiques**: Les caractéristiques extraites peuvent ne pas capturer toutes les nuances des fichiers malveillants. L'exploration de caractéristiques supplémentaires ou l'utilisation de techniques avancées d'extraction de caractéristiques peuvent améliorer les performances.

- **Tuning des Hyperparamètres**: Certains modèles peuvent nécessiter un ajustement fin des hyperparamètres pour optimiser leurs performances. Une recherche systématique des hyperparamètres peut être entreprise.

Conclusion

Le projet de Détection de Malware a abouti à la création d'un pipeline complet, de l'importation du dataset au déploiement du modèle. Les résultats obtenus montrent que le modèle Random Forest a la meilleure précision parmi les modèles testés. L'ensemble du processus met en lumière l'importance du choix des caractéristiques, de la taille du dataset, et de l'optimisation des hyperparamètres dans le développement de modèles de machine learning efficaces pour la détection de malware.