Projet SecureNLP
Cybersécurité et IA

Ce projet ancre vos connaissances :

- **Sécurité appliquée** (authentification, chiffrement, droits d'accès, journalisation…)
- **Pipeline NLP** (prétraitement, entraînement, prédiction sur un jeu de données texte)

Vous êtes une équipe de 4 data scientists en cybersécurité, chargée de mettre en place un pipeline de machine learning pour classer des commentaires en ligne (positifs / négatifs), tout en assurant la **sécurité des données**, du modèle et de l'environnement d'exécution.

Contexte du projet :

L'entreprise **"SentimentCorp"** développe une application de modération automatique de commentaires. Le projet pilote consiste à entraîner un modèle NLP pour détecter les commentaires toxiques ou négatifs à partir d'un jeu de données texte.

Cependant, le responsable de la sécurité vous rappelle que :

- Les commentaires proviennent de **sources sensibles** (ex. forums internes d'employés, clients…).
- Le modèle doit être **hébergé en interne**, avec une **traçabilité** des accès.
- Les données d'entraînement **ne doivent pas fuiter**.
- Le pipeline doit être **auditable** (journalisation), et respecter le **principe du moindre privilège**.

Données fournies :

Tâches à réaliser :

Partie 1 – Pipeline NLP classique

1. Charger le dataset.
2. Nettoyer les données (lowercase, suppression des stop words, lemmatisation).
3. Transformer le texte avec TF-IDF.
4. Entraîner un modèle (Logistic Regression ou autre).
5. Évaluer avec précision, rappel, f1-score.

Partie 2 – Sécurisation du pipeline

a. Sécurisation des données

- Implémentez un chiffrement simple (ex : Fernet, AES ou un hash des IDs utilisateurs).
- Démontrer comment empêcher la réidentification.
- Supprimez les colonnes inutiles avant export.

b. Gestion des accès

- Implémentez un système d'accès aux données basé sur des rôles simulés :
  - Data Scientist : accès complet
  - Analyste : accès aux prédictions seulement
- Simulez la séparation via des notebooks ou des scripts distincts.

c. Journalisation des accès

- Créez un log simple (log_access.txt) :
  - Quand quelqu'un charge les données
  - Quand une prédiction est faite

d. Protection du modèle

- Enregistrez le modèle avec une **signature ou un hash**.
- Montrez comment détecter s'il a été modifié.

Bonus !

- Ajoutez une **authentification par mot de passe** ou **token simulé** dans le script.
- Intégrez une simple **interface Streamlit sécurisée**.
- Proposez un mécanisme de **chiffrement des prédictions envoyées à l'utilisateur**.

Livrables attendus

- Un ou plusieurs notebooks Python organisés
- Un fichier README.md expliquant votre pipeline et les choix de sécurité
- Un fichier log_access.txt simulant la journalisation
- Le modèle enregistré et vérifié