

Projet de Détection de Spam SMS

Application Web Interactive avec Streamlit

Objectif du projet: Développer une application web interactive pour détecter les messages SMS de type spam.

Technologies utilisées: Streamlit, Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn, TF-IDF Vectorizer.



Chargement des Données + Nettoyage des Messages

Chargement des Données

Utilisation de `pd.read_csv` pour lire les fichiers CSV contenant les messages SMS étiquetés.

Exemple de chemin de fichier: "C:\Users\sbond\Desktop\SPAM Sms detection\SMSSpamCollection.csv"

Nettoyage des Messages

Suppression des mots vides (stopwords) et conversion en minuscules.

Fonction `clean_message` pour nettoyer chaque message.

Entraînement du Modèle - Partie 1

1

Vectorisation

Vectorisation des messages avec TfidfVectorizer.

Transformation des messages textuels en vecteurs numériques.

2

Classification

Utilisation de la régression logistique (LogisticRegression) pour classer les messages.

3

Séparation des Données

Séparation des données en ensembles d'entraînement et de test.



Évaluation du Modèle - Métriques

```
# Diviser en ensembles d'entraînement et de test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Définir les modèles
models = {
    "Régression Logistique": LogisticRegression(max_iter=500),
    "Arbre de Décision": DecisionTreeClassifier(),
    "Forêt Aléatoire": RandomForestClassifier(),
    "SVM": SVC(probability=True),
    "k-NN": KNeighborsClassifier(),
    "Naive Bayes": GaussianNB(),
}

# Initialiser un tableau de résultats
results = {}

# Entraîner et évaluer chaque modèle
for name, model in models.items():
    if name == "Naive Bayes": # Naive Bayes ne supporte pas directement les
        model.fit(X_train.toarray(), y_train)
        y_pred = model.predict(X_test.toarray())
        y_prob = None # GaussianNB ne fournit pas de méthode predict_proba
    else:
        model.fit(X_train, y_train)
        y_pred = model.predict(X_test)
        y_prob = model.predict_proba(X_test)[:, 1] if hasattr(model, "predict_proba") else None

    accuracy = accuracy_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred, average='weighted')
    recall = recall_score(y_test, y_pred, average='weighted')
    auc = roc_auc_score(y_test, y_prob) if y_prob is not None else np.nan
```

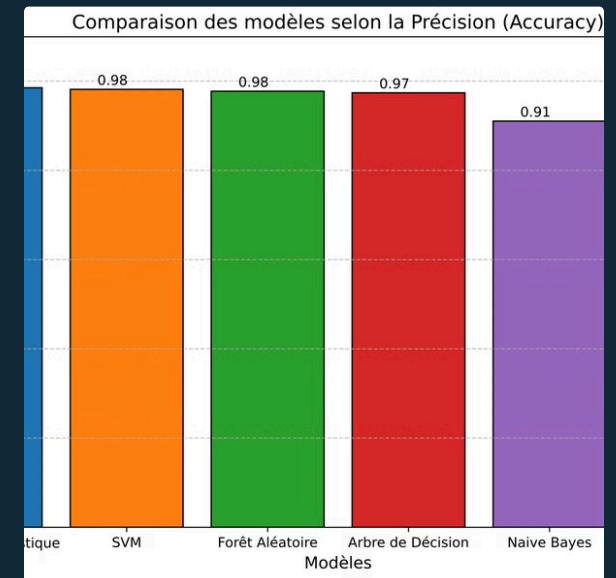
Definition des modèles

Regressions Logique VS Arbre de décision
VS Forêt aléatoire VS SVM vs K-NN VS
Naive bayes

```
metrics_titles = {
    "Précision": "Précision",
    "Score F1": "Score F1",
    "Rappel": "Rappel",
    "AUC": "AUC (Surface)"
}
```

Definition des Metrics

Évaluation des performances du modèle
avec confusion_matrix.



Résultats Clés

Affichage des comparatifs clés: précision,
rappel, F1-score.

Visualisation des Résultats



Distribution des Classes

Graphique en barres montrant la répartition des messages entre "HAM" et "SPAM".



Matrice de Confusion

Heatmap indiquant les performances du modèle en termes de vrais positifs, faux positifs, vrais négatifs et faux négatifs.



Courbe Précision-Rappel

Courbe montrant la trade-off entre précision et rappel.

Interaction Utilisateur - Test d'un SMS Unique

1

Entrée du SMS

Entrée manuelle d'un message SMS pour obtenir une prédiction en temps réel.

2

Nettoyage et Vectorisation

Le SMS est nettoyé et vectorisé pour l'analyse.

3

Prédiction

Affichage de la prédiction et de la probabilité associée.

Interaction Utilisateur - Téléchargement de Nouvelles Données

1

Téléchargement

Téléchargement d'une nouvelle base de données pour enrichir l'ensemble de formation.

2

Fusion

Fusion des nouvelles données avec l'ensemble existant.

3

Réentraînement

Réentraînement du modèle avec les données combinées.



Thank you! Please remember:
In your an SMS spam is for confortitude.

Conclusion + Questions

Résumé du Projet

Solution pratique et interactive pour la détection de spam SMS.

Amélioration Continue

Possibilité de mettre à jour le modèle avec de nouvelles données pour améliorer ses performances.

Techniques Avancées

Combinaison de techniques avancées de NLP et de machine learning.

Merci pour votre attention Nous sommes disponible pour répondre à vos questions.