CLASSIFICATION DES MAILS

Auteurs

Baye Cheikh Mbaye

Harry Phillips

Mohamed-Vall Mohamedou

Moussa Konate

Professeur

Corentin Vasseur

Table des matières

Introduction

Dans le monde numérique d'aujourd'hui, la gestion efficace des communications électroniques est cruciale. Parmi les milliards d'emails échangés quotidiennement, la distinction entre messages légitimes et indésirables, tels que le spam, devient une priorité majeure. La présente étude se penche sur cette problématique en explorant les techniques avancées de classification des emails.

Le jeu de données à notre disposition offre une opportunité unique de plonger dans l'univers complexe des communications électroniques. Constitué d'une variété de messages, notre objectif est de développer un modèle robuste de Machine Learning capable de discerner avec précision entre les mails considérés comme légitimes (ham) et ceux indésirables (spam). Pour atteindre cet objectif, nous avons opté pour l'utilisation d'un modèle Support Vector Machine (SVM), reconnu pour son efficacité dans la classification de texte.

Au fil de ce rapport, nous explorerons les différentes étapes du processus, de la préparation du jeu de données à la mise en œuvre du modèle SVM. Nous analyserons également les performances du modèle en utilisant des métriques clés telles que l’accuracy, le recall et le F1-score. Cette approche méthodique nous permettra de mieux comprendre la capacité du modèle à généraliser et à identifier les emails indésirables avec précision.

En somme, cette étude se veut une exploration approfondie de la classification des emails, mettant en lumière l'efficacité d'un modèle SVM dans la résolution de cette tâche cruciale de filtrage du contenu électronique.

Prétraitement

Avant de faire d'entamer l'étude de nos données, nous avons fait un prétraitement pour mieux nettoyer nos données afin de ne garder que les données et variables utiles pour notre étude. Pour ce faire, nous avons utilisé un programme python qui utilise la bibliothèque NLTK (Natural Language Toolkit) et scikit-learn pour prétraiter des données textuelles, probablement dans le contexte de l'analyse d'emails.

Voici ce que fait le code, étape par étape (Code à retrouver sur Git):

1. Importe les modules nécessaires, notamment nltk, re (expressions régulières), et certaines fonctions de scikit-learn pour le traitement des données.
2. Télécharge les ressources linguistiques nécessaires à NLTK, telles que le tokenizer punkt et la liste des stopwords.
3. Définit une fonction **clean_text** qui prend en entrée un texte (dans notre cas un email) et effectue plusieurs opérations de nettoyage, la conversion en minuscules, la suppression de la ponctuation, des chiffres, et des mots vides (stopwords).
4. Applique la fonction **clean_text** à la colonne 'text' d'un ensemble de données (probablement un DataFrame), et stocke le texte nettoyé dans une nouvelle colonne appelée 'cleaned_text'.

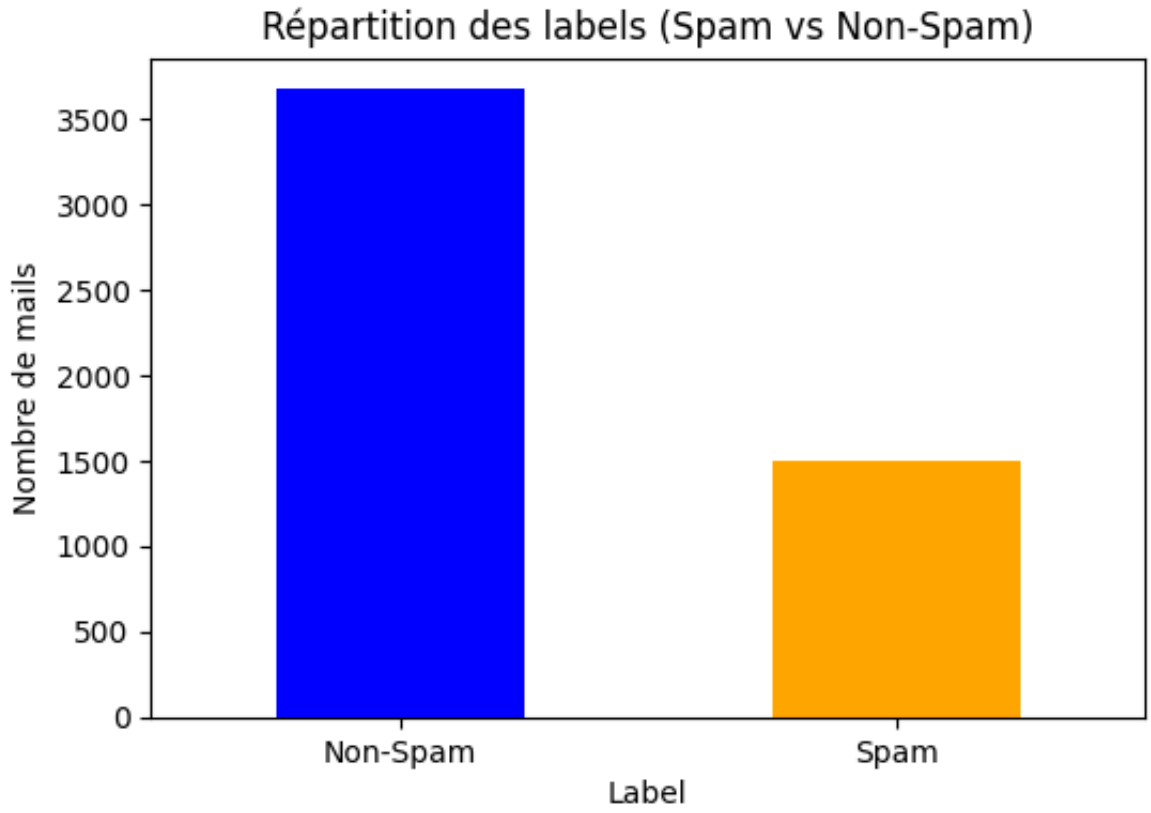
Quant à ta question sur le "ham" dans un email, le terme "ham" est dans notre cas représente les mails indésirables. En d'autres termes, un "ham" est un courrier électronique que l'utilisateur souhaite recevoir, par opposition aux "spams" qui sont des courriers indésirables. En résumé, notre code dans cette étape on nettoie le texte des emails en le préparant pour une analyse ultérieure dans le but de classer les emails en tant que spam ou non-spam en fonction de leur contenu.

Description du fichier et Statistiques descriptives

Notre jeu de données est composé de 5171 observations pour 4 variables.

Le fichier contient un message par ligne. Chaque ligne est composée de deux colonnes :une colonne contient l'étiquette (ham ou spam) et et l'autre contient le texte brut.

Nous allons faire quelques statistiques descriptives, ces statistiques auront pour but de croiser nos variables pour voir des informations importantes qui nous permettront de mieux exploiter nos données et de bien aborder la modélisation.



Sur le graphique ci-dessus, nous remarquons qu'il y a beaucoup plus de mails non-spam que de mails spam. Nous avons un peu plus de 3500 mails non-spam pour environ
1500 mails spam.
Ce constat est très important car nous remarquons déjà qu'il y a un déséquilibre au niveau de nos modalités. Nous allons prendre en compte cette remarque dans la modélisation pour faire un modèle qui tiendra compte de ce déséquilibre afin de nous faire un meilleur modèle.

Modélisation et choix du modèle

Dans cette partie nous allons construire notre modèle. En effet l'intérêt de cette partie est de faire un modèle qui pourra bien s'adapter à nos données afin de faire de meilleures prédictions.
Dans notre cas, nous avons choisi le modèle SVM, ce choix s'explique par plusieurs raisons parmi lesquelles nous avons:

- **Robustesse aux problèmes de surajustement :** Les SVM ont des mécanismes intégrés pour éviter le surajustement (overfitting), ce qui signifie qu'ils peuvent généraliser bien même avec des ensembles de données limités.
- **Marges maximales :** Les SVM cherchent à maximiser la marge entre les différentes classes, ce qui conduit souvent à de meilleures performances de généralisation sur de nouveaux exemples.
- **Bonnes performance**s pour les tâches de classification binaire : Les SVM sont particulièrement bien adaptés aux problèmes de classification binaire, où l'objectif est de séparer les exemples de deux classes différentes.

Pour mener à bien cette étude, nous avons partagé notre jeu de données en deux ensemble: Un ensemble d'apprentissage (80%) et un ensemble de tests (20%).

Découper un jeu de données en deux ensembles distincts, un jeu d'apprentissage et un jeu de test est essentiel. L'ensemble d'apprentissage est utilisé pour entraîner le modèle, tandis que l'ensemble de test sert à évaluer objectivement ses performances sur de nouvelles données. Cette pratique prévient le surajustement du modèle aux données d'entraînement et offre une estimation non biaisée de sa capacité à généraliser. En garantissant que le modèle peut traiter des données inconnues, cette approche renforce la fiabilité des résultats et favorise le développement de modèles plus robustes et généralisables.

Reste encore à définir les métriques de performances que nous prendrons en considération lors de la comparaison des modèles.

Performance du modèle

Comme dit auparavant, nous avons fait le modèle d'entraînement et ensuite nous avons fait le testé le modèle, nous avons eu les résultats suivants:

Accuracy	Recall	f1-score
0,98	0,99	0,99

Accuracy:

Notre accuracy vaut 0.98 ce qui veut dire que la proportion de prédictions correctes parmi toutes les prédictions effectuées par le modèle est de 0.98.

Recall:

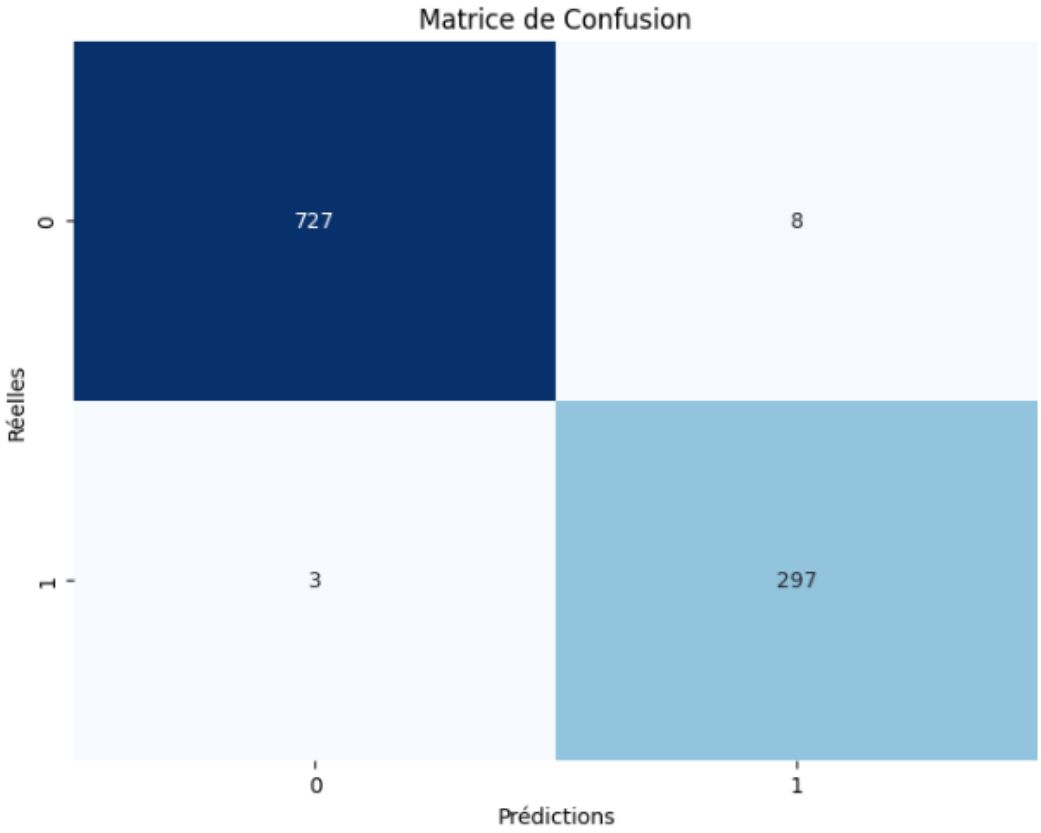
Le recall mesure la capacité d'un modèle à identifier tous les exemples positifs réels, ici nous avons un recall de 0.99 ce qui veut dire que notre modèle a une bonne capacité d'identifier les exemples positifs réels.

F1-score:

Il est très utile lorsque les classes sont déséquilibrées car il prend en compte les erreurs liées aux faux positifs et aux faux négatifs. Dans notre cas nous avons un F1-score de 0.99 ce qui témoigne de la bonne performance du modèle à bien classer les mails.

Matrice de confusion:

Cette matrice nous permet de voir la comparaison entre nos prédictions et les valeurs réelles .
On peut voir sur notre matrice que nos prédictions et valeurs réelles sont bonnes.



Attention !

Nous allons nous baser sur le F1-score pour juger la qualité de notre modèle, Le F1-score fournit une mesure unique qui résume la performance d'un modèle, ce qui peut être plus facile à communiquer et à interpréter que plusieurs métriques individuelles.Il particulièrement utile dans notre contexte où la simplicité et la clarté sont prioritaires.

Conclusion

En conclusion, cette étude sur la classification des emails à l'aide d'un modèle Support Vector Machine (SVM) a dévoilé des perspectives prometteuses dans la lutte contre le fléau du spam électronique. L'analyse approfondie du jeu de données et l'application judicieuse du modèle SVM ont permis d'obtenir des résultats significatifs, soulignant la pertinence de cette approche pour la détection précise des messages indésirables.

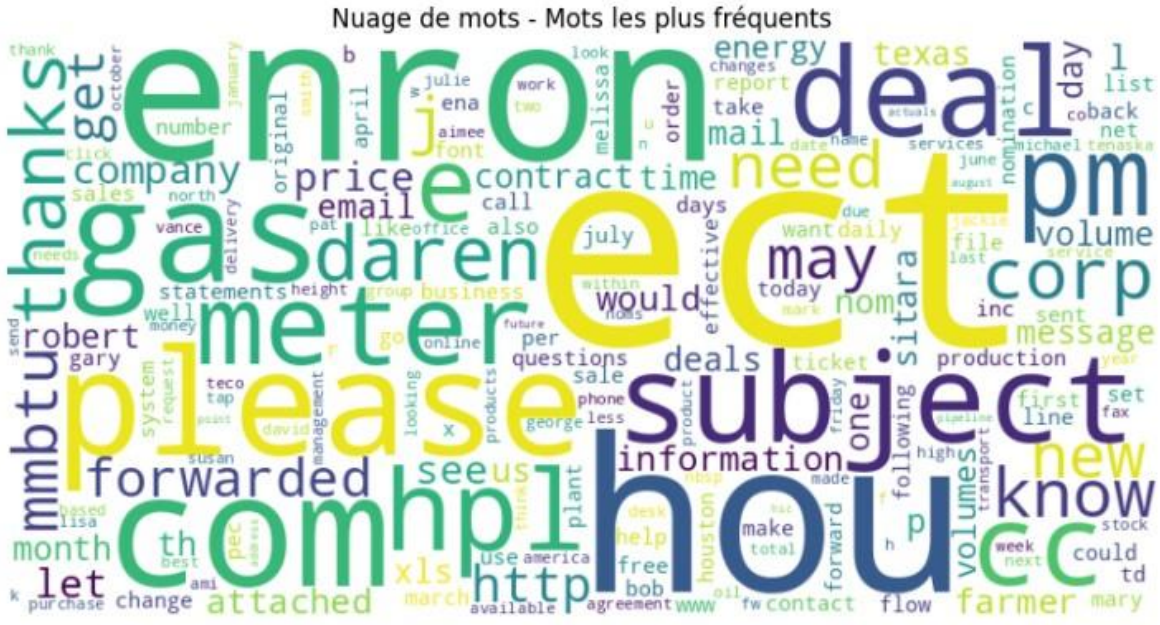
Les performances du modèle, évaluées à travers des métriques telles que la précision, le rappel et le F1-score, témoignent de sa capacité à équilibrer avec succès la minimisation des faux positifs et des faux négatifs. Cette capacité est cruciale dans un contexte où la préservation de la confidentialité des utilisateurs et la réduction des interruptions causées par le spam sont des impératifs.

Il convient de noter que, bien que le modèle SVM ait démontré son efficacité, la lutte contre le spam est un défi continu. Les évolutions constantes des techniques de spamming exigent une adaptation continue des méthodes de filtrage. Les travaux futurs pourraient explorer des approches d'apprentissage automatique plus avancées et intégrer des mécanismes de mise à jour dynamique pour rester à la pointe de la détection du spam.

En définitive, cette étude souligne l'importance cruciale de la classification des emails dans le contexte de la communication électronique moderne. En combinant la puissance du modèle SVM avec une approche méthodique, nous avons posé les bases d'une défense robuste contre les mails indésirables dans nos boîtes de réception, contribuant ainsi à une expérience utilisateur plus sécurisée et efficace.

Annexes

Les mots les plus fréquents



Lien du jeu de données

https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset/data