ĐHQGHN
Since 1906
VNU
ĐẠI HỌC QUỐC GIA HÀ NỘI
Vietnam National University, Hanoi

ALGORITHMES DE CLASSIFICATION

Algorithmes d'apprentissage automatique pour la classification des courriers indésirables

Présenté par groupe 12

- **ADOUM Okim Boka**

Prof: Lê Hồng Phương

Plan

1. Introduction
2. Outils
3. Naive Bayes
4. RNN
5. Random Forest
6. Comparaison
7. Conclusion.

Introduction

Le volume croissant de courrier électronique en vrac non sollicité (également appelé spam) a généré un besoin de filtres anti-spam fiables. Les techniques d'apprentissage automatique sont désormais utilisées depuis plusieurs jours pour filtrer automatiquement le courrier indésirable à un taux très correct. En ce qui concerne notre travail, nous passons en revue certaines des méthodes d’apprentissage automatique les plus populaires (classification bayésienne, k-NN, random forest) et de leur applicabilité au problème de la classification du courrier indésirable par courrier électronique.

Outils

Environnement de développement et langage de programmation

- **google colaboratory**
- **Python 3.6**

Algorithmes

- Naive Bayes
- Random Forest
- RNN

Dataset

spam de kaggle

Preparation de données

Nous avons utilisé un jeu de données de 5572 enregistrements.

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

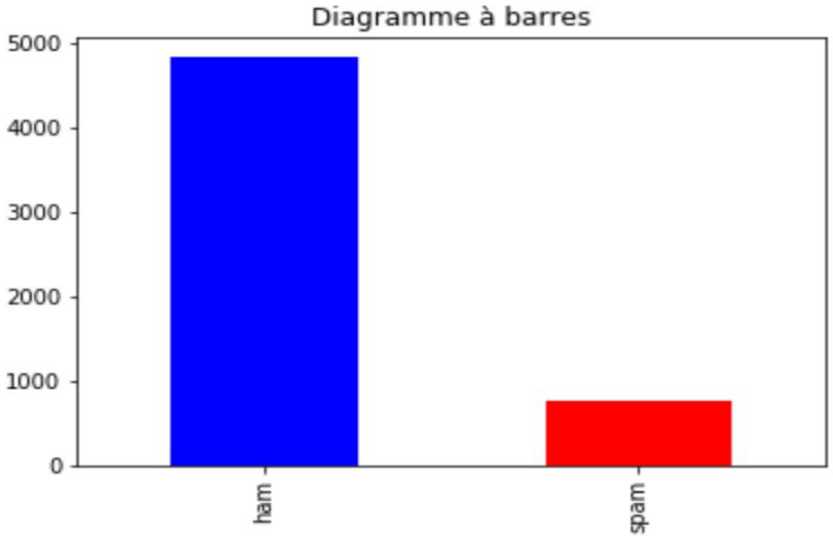
Tokenisation (transformation des mots en vecteurs) des mots (affectation d'identifiant au mot),
Determiner la frequence de chaque mot du dataset

transformer la variable spam / non-spam en variable binaire

Division du dataset (77% pour le training et 33% pour le test)

On a 8404 nouvelles features

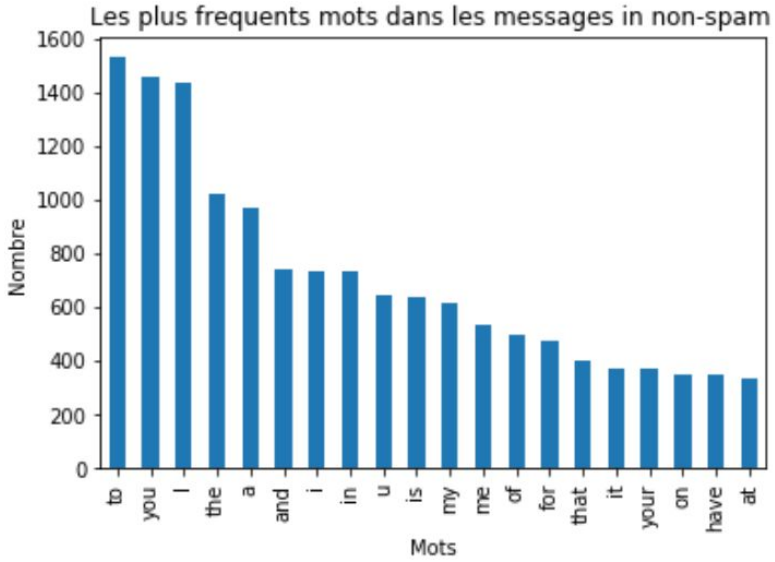
C'est ce jeu de donnée que nous allons pour le naive bayes et Le random forest



Distribution de graphiques de spam / non-spam

Preparation de données

Les mots les plus fréquents dans un spam et non spam



Naive Bayes

Naive Bayes Classifier est un algorithme populaire en Machine Learning. C'est un algorithme du Supervised Learning utilisé pour la classification. Il est particulièrement utile pour les problématiques de classification de texte. Un exemple d'utilisation du Naive Bayes est celui du filtre anti-spam.

Naive Bayes /Resultats

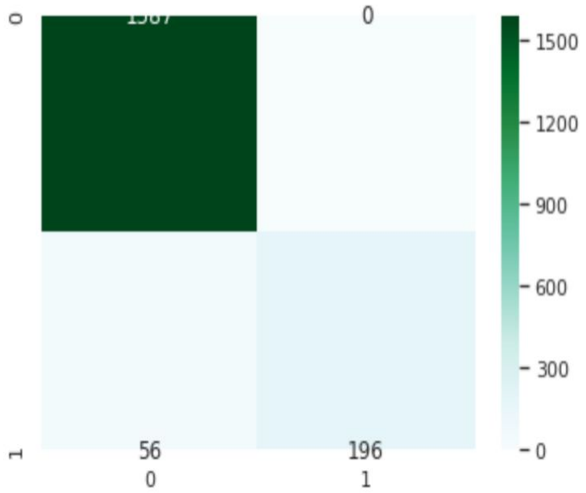
Résultats

```
best_index = models['Test Precision'].idxmax()
models.iloc[best_index, :]
```

```
alpha              15.730010
Train Accuracy      0.979641
Test Accuracy       0.969549
Test Recall         0.777778
f1_score_test       0.875000
Test Precision      1.000000
Name: 143, dtype: float64
```

Matrice de confusion

```
ax=sns.heatmap(m_confusion_test, annot=True, fmt="d",cmap='BuGn')
```



	Predicted 0	Predicted 1
Actual 0	1587	0
Actual 1	56	196

Random Forest /Résultats

Résultat

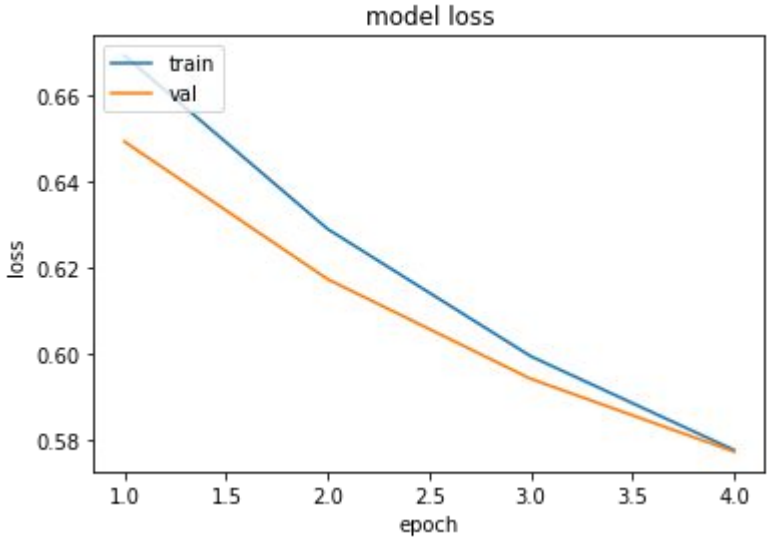
```
Score Training
[0.8716849718724886, 0.8775783552102866, 0.8845432627913207, 0.9129386552370747, 0.9373158317706938, 0.9450843825341548, 0.9504420037503348]
Test Training
[0.8656878738444806, 0.8743882544861338, 0.8809135399673735, 0.9048395867319196, 0.9287656334964655, 0.9358346927678086, 0.9390973355084284]
```

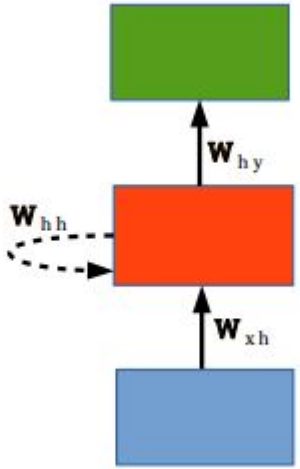
Résultat avec le critère Entropie

```
Accuracy score: 0.9744426318651441
Precision score: 1.0
Recall score: 0.8134920634920635
F1 score: 0.897155361050328 2
```

Réseau de neurone récurrent

```
Test de précision: 0.7740
```





Représentation compacte des RNN. Toutes les flèches représentent des connexions complètes. La flèche en pointillée représente les connexions ayant un décalage temporel (t − 1)

Réseau de neurone récurrent / Dataset

Nous avons utilisé trois dataset

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

https://www.kaggle.com/karthickveerakumar/spam-filter

https://www.kaggle.com/venky73/spam-mails-dataset

https://www.kaggle.com/ozlerhakan/spam-or-not-spam-dataset

Comparaison

Algorithmes	
Naive Bayes	0.969549
Random Forest	0.9782490483958673
RNN	0.7740

Note

- **ESSO Dissirama -> 9**
- **ADOUM Okim Boka -> 9**