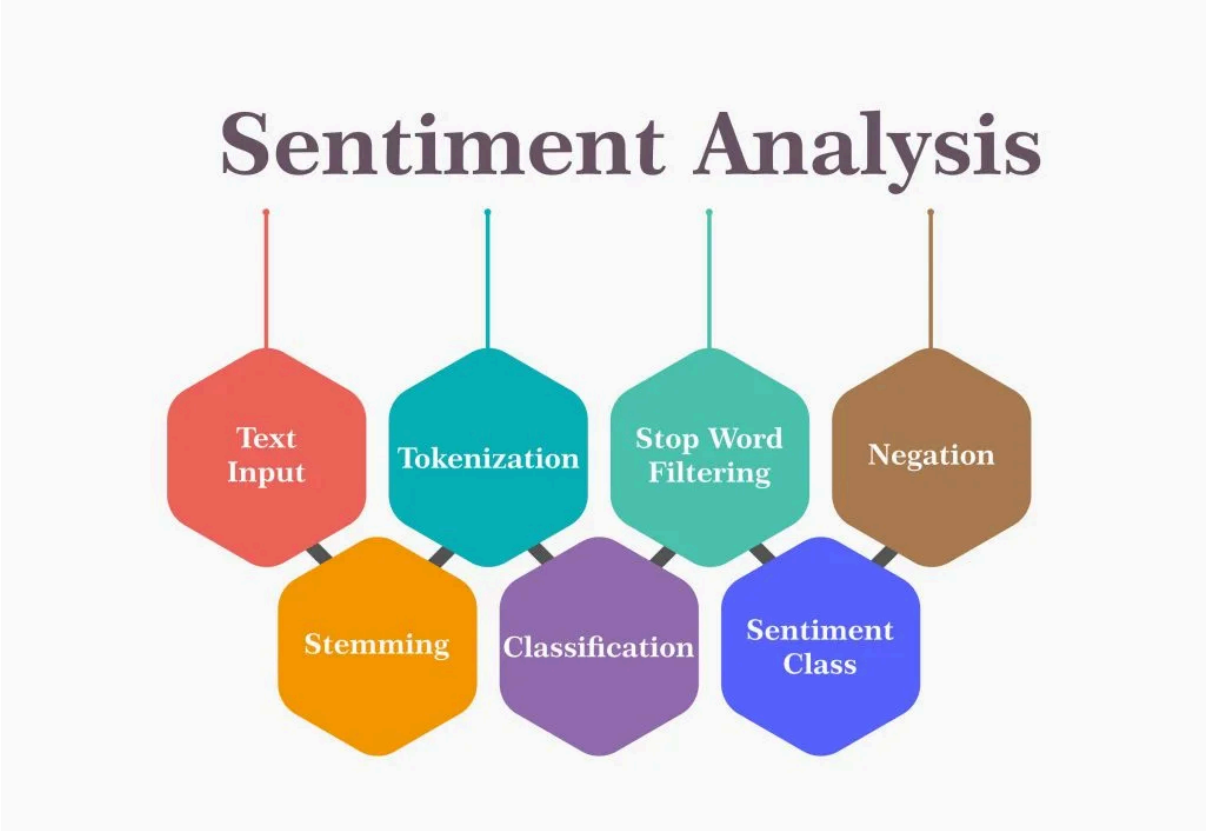
Rapport d’analyse de sentiment sur Twitter



Réalisé par :

- Jalakshana KANNAN

1.Introduction

Ce projet consiste à développer un modèle de classification pour déterminer le sentiment (4 = positif ou 0 = négatif) des tweets en utilisant l'ensemble de données Sentiment140. L'objectif est de comparer plusieurs modèles d'apprentissage automatique appliqués à des échantillons de différentes tailles.

Il contient 1 600 000 tweets avec des colonnes suivants :

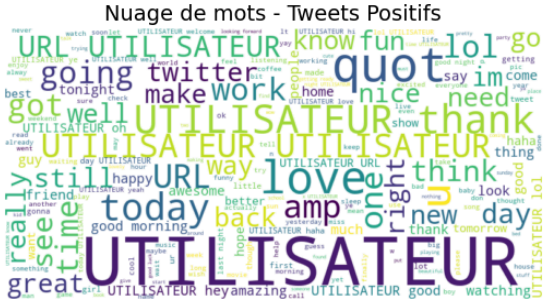
- **sentiment** : La polarité du tweet (0 = négatif, 4 = positif).
- **ids** : L'identifiant du tweet (2087).
- **date** : La date du tweet (Sat May 16 23:58:44 UTC 2009).
- **flag** : La requête (lyx). S'il n'y a pas de requête, cette valeur est NO_QUERY.
- **user** : l'utilisateur qui a tweeté (robotickilldozr).
- **text** : le texte du tweet (Lyx est cool).

2.Méthodologie utilisée :

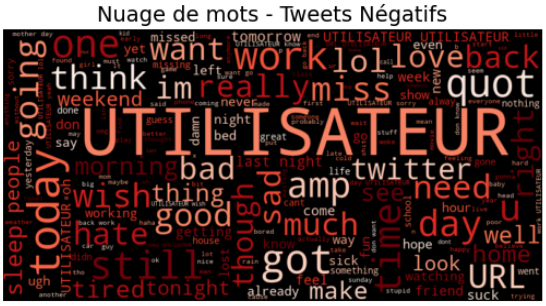
Prétraitement des données

1. **Chargement et nettoyage des tweets** :
    - On supprime les **URLs et mentions (@)**.
    - On remplace les **emojis** par une description textuelle.
    - On supprime les **caractères spéciaux** et conversion en **minuscules**.
    - On procède à la **lemmatisation** pour réduire les mots à leur forme de base.
2. **Visualisation des données** :
    - **Nuages de mots** pour les tweets **positifs** et **négatifs**.

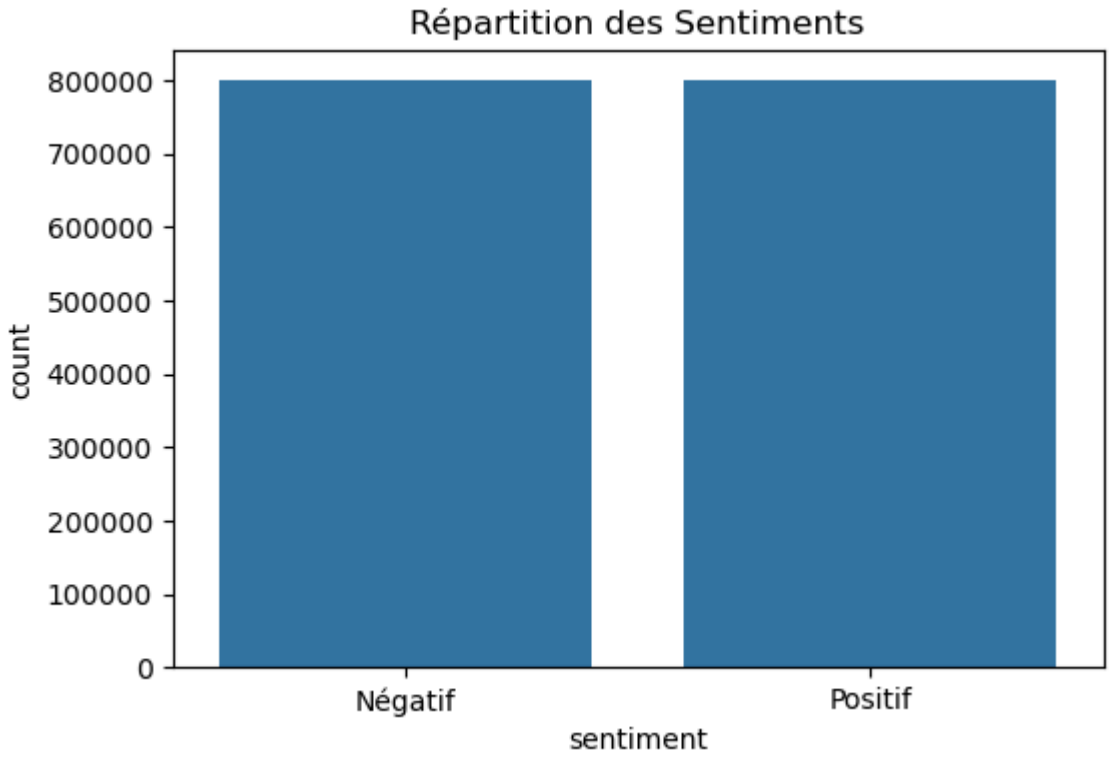
Nuage de mots - Tweets Positifs



Nuage de mots - Tweets Négatifs



- **Distribution des sentiments** avec un histogramme.



3.Modèles évalués

1. Bernoulli Naive Bayes
2. Support Vector Classifier (SVM)
3. Régression Logistique
4. Random Forest
5. XGBoost

4.Métriques d'évaluation

- Matrices de confusion.
- Précision, Rappel, F1-score pour chaque classe et moyenne pondérée.
- Comparaison entre échantillons de tailles différentes (800 000, 900 000 et 1 000 000 tweets).

5.Résultats obtenus

Échantillon : 800 000 tweets

Modèle	**Précision**
Bernoulli Naive Bayes	76,46%
Support Vector Classifier	76,68%
Régression Logistique	77,46%
Random Forest	69,83%

XGBoost	73,25%

```
*** Bernoulli Naive Bayes - Échantillon 800000 ***
Précision : 0.7646
Rapport de classification :
              precision    recall  f1-score   support

           0       0.77      0.76      0.76     79976
           1       0.76      0.77      0.77     80024

    accuracy                           0.76    160000
   macro avg       0.76      0.76      0.76    160000
weighted avg       0.76      0.76      0.76    160000
```

```
*** Support Vector Classifier - Échantillon 800000 ***
c:\Users\jalak\anaconda3\Lib\site-packages\sklearn\svm\
  warnings.warn(
Précision : 0.7668
Rapport de classification :
              precision    recall  f1-score   support

           0       0.78      0.75      0.76     79976
           1       0.76      0.78      0.77     80024

    accuracy                           0.77    160000
   macro avg       0.77      0.77      0.77    160000
weighted avg       0.77      0.77      0.77    160000
```

```
*** Régression Logistique - Échantillon 800000 ***
Précision : 0.7746
Rapport de classification :
              precision    recall  f1-score   support

           0       0.78      0.76      0.77     79976
           1       0.77      0.79      0.78     80024

    accuracy                           0.77    160000
   macro avg       0.78      0.77      0.77    160000
weighted avg       0.78      0.77      0.77    160000
```

```
*** Random Forest - Échantillon 800000 ***
Précision : 0.6983
Rapport de classification :
              precision    recall  f1-score   support

           0       0.70      0.70      0.70     79976
           1       0.70      0.69      0.70     80024

    accuracy                           0.70    160000
   macro avg       0.70      0.70      0.70    160000
weighted avg       0.70      0.70      0.70    160000
```

```
*** XGBoost - Échantillon 800000 ***
c:\Users\jalak\anaconda3\Lib\site-packages\xgboost\core
Parameters: { "use_label_encoder" } are not used.

  warnings.warn(smsg, UserWarning)
Précision : 0.7325
Rapport de classification :
              precision    recall  f1-score   support

           0       0.73      0.73      0.73     79976
           1       0.73      0.74      0.73     80024

    accuracy                           0.73    160000
   macro avg       0.73      0.73      0.73    160000
weighted avg       0.73      0.73      0.73    160000
```

Échantillon : 900 000 tweets

Modèle	**Précision**
Bernoulli Naive Bayes	76,42%
Support Vector Classifier	76,75%
Régression Logistique	77,44%
Random Forest	66,85%
XGBoost	73,23%

```
### Entraînement sur un échantillon de 900000 tweets ###

*** Bernoulli Naive Bayes - Échantillon 900000 ***
Précision : 0.7642
Rapport de classification :
              precision    recall  f1-score   support

           0       0.77      0.76      0.76     89991
           1       0.76      0.77      0.77     90009

    accuracy                           0.76    180000
   macro avg       0.76      0.76      0.76    180000
weighted avg       0.76      0.76      0.76    180000
```

```
*** Support Vector Classifier - Échantillon 900000 ***
c:\Users\jalak\anaconda3\Lib\site-packages\sklearn\svm\
  warnings.warn(
Précision : 0.7675
Rapport de classification :
              precision    recall  f1-score   support

           0       0.78      0.75      0.76     89991
           1       0.76      0.79      0.77     90009

    accuracy                           0.77    180000
   macro avg       0.77      0.77      0.77    180000
weighted avg       0.77      0.77      0.77    180000
```

```
*** Régression Logistique - Échantillon 900000 ***
Précision : 0.7744
Rapport de classification :
              precision    recall  f1-score   support

           0       0.79      0.75      0.77     89991
           1       0.76      0.79      0.78     90009

    accuracy                           0.77    180000
   macro avg       0.77      0.77      0.77    180000
weighted avg       0.77      0.77      0.77    180000
```

```
*** Random Forest - Échantillon 900000 ***
Précision : 0.6685
Rapport de classification :
              precision    recall  f1-score   support

           0       0.67      0.67      0.67     89991
           1       0.67      0.67      0.67     90009

    accuracy                           0.67    180000
   macro avg       0.67      0.67      0.67    180000
weighted avg       0.67      0.67      0.67    180000
```

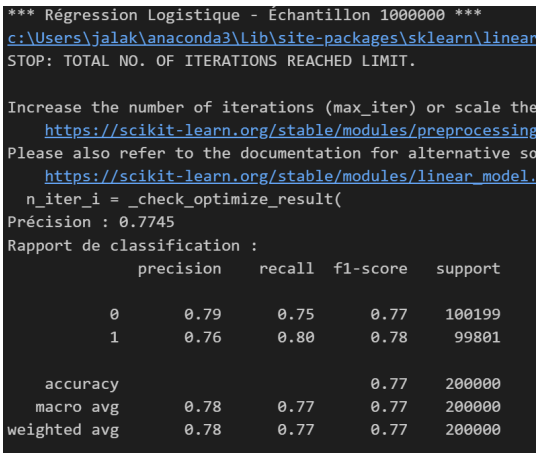
```
*** XGBoost - Échantillon 900000 ***
c:\Users\jalak\anaconda3\Lib\site-packages\xgboost\cor
Parameters: { "use_label_encoder" } are not used.

  warnings.warn(smsg, UserWarning)
Précision : 0.7323
Rapport de classification :
              precision    recall  f1-score   support

           0       0.73      0.73      0.73     89991
           1       0.73      0.74      0.73     90009

    accuracy                           0.73    180000
   macro avg       0.73      0.73      0.73    180000
weighted avg       0.73      0.73      0.73    180000
```

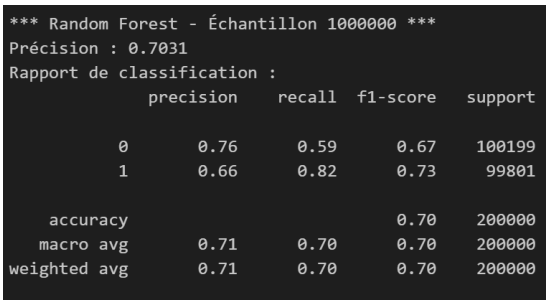
Échantillon : 1 000 000 tweets

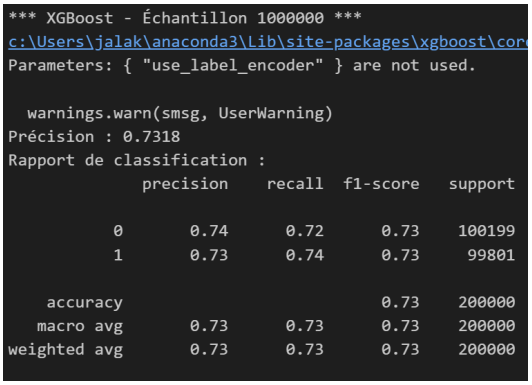
Modèle	**Précision**
Bernoulli Naive Bayes	76,41%
Support Vector Classifier	76,87%
Régression Logistique	77,45%
Random Forest	70,31%
XGBoost	73,18%

```
### Entraînement sur un échantillon de 1000000 tweets ###

*** Bernoulli Naive Bayes - Échantillon 1000000 ***
Précision : 0.7641
Rapport de classification :
              precision    recall  f1-score   support

           0       0.77      0.76      0.76    100199
           1       0.76      0.77      0.77     99801

    accuracy                           0.76    200000
   macro avg       0.76      0.76      0.76    200000
weighted avg       0.76      0.76      0.76    200000
```

```
*** Support Vector Classifier - Échantillon 1000000 ***
c:\Users\jalak\anaconda3\Lib\site-packages\sklearn\svm\_
  warnings.warn(
Précision : 0.7687
Rapport de classification :
              precision    recall  f1-score   support

           0       0.78      0.75      0.76    100199
           1       0.76      0.79      0.77     99801

    accuracy                           0.77    200000
   macro avg       0.77      0.77      0.77    200000
weighted avg       0.77      0.77      0.77    200000
```







6.Défis rencontrés

Qualité des données : De nombreux tweets contenaient des éléments non pertinents tels que des URL, des émojis et des mentions inutiles, qui nécessitaient un prétraitement pour améliorer la qualité.

Complexité linguistique : Les tweets, souvent courts, contenaient des abréviations, de l'argot, des fautes d'orthographe et de l'ironie, ce qui rendait difficile de saisir le véritable sentiment exprimé dans le texte.

Temps d'exécution : L'entraînement de certains modèles, en particulier SVC ou Random Forest, a été long en raison de la taille des données.

6.Conclusion et perspectives

Les résultats obtenus confirment que la Régression Logistique est le modèle le plus performant pour la classification des sentiments des tweets, avec une précision et un F1-score atteignant 77.45%. Elle surpasse légèrement le Support Vector Classifier, qui reste néanmoins une alternative robuste. Ce modèle se distingue par sa capacité à bien équilibrer les classes, tout en maintenant une bonne généralisation sur des ensembles de données de différentes tailles.

À l'inverse, Bernoulli Naive Bayes, bien que rapide et stable avec une précision de 76%, montre des limites dans la gestion des faux positifs et négatifs. XGBoost, avec 73% de précision, offre des performances respectables mais nécessite un ajustement plus précis des hyperparamètres. Enfin, Random Forest, avec une précision avoisinant 70%, apparaît

comme le modèle le moins adapté à cette tâche, notamment en raison d’un taux élevé d’erreurs de classification.

L'augmentation de la taille de l’échantillon n’a pas entraîné d’amélioration significative des performances, suggérant une saturation des informations exploitables par ces modèles. Pour aller plus loin, il serait pertinent d’explorer des modèles de deep learning comme BERT ou GPT, capables de mieux saisir le contexte et les subtilités linguistiques des tweets. Une analyse multi-labels pourrait également être intégrée pour inclure des sentiments plus nuancés (neutre, mixte), offrant ainsi une classification plus fine et plus adaptée aux réalités du langage naturel.