

# Détecteur d'insultes Twitter

Antoine Dieu, Etienne Lefranc, Ambroise Marché, Matthieu Scharffe, Quentin  
Echasseriau, Lisa Lupi – Coding weeks 2021

[lien vers le dépôt GitLab](#)

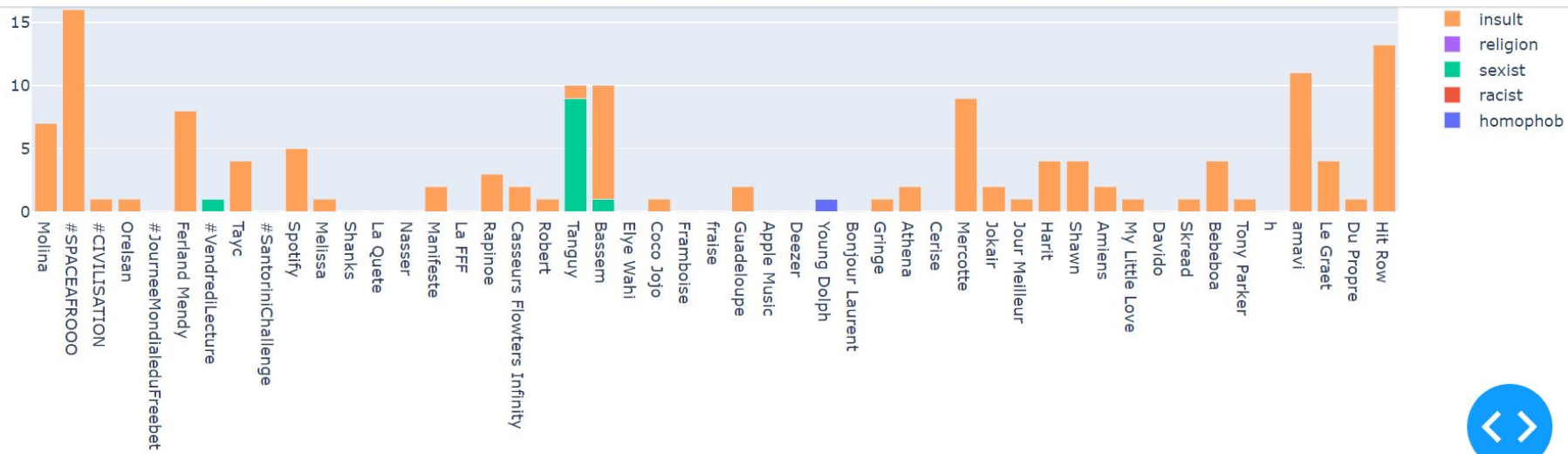


Paris

Paris

London

New York



## Besoins et usages

Déterminer la fréquence des insultes dans les tendances, qui représentent les sujets les plus discutés sur Twitter afin d'évaluer lesquels sont les plus polémiques.





**MVP**

Afficher les notes des tendances  
françaises sur dash sous forme  
d'histogramme



# Découpage du projet

## **Sprint 1 – Collecte et stockage de données**

Fonctionnalité 1 : Collecte tendances

Fonctionnalité 2 : Collecte tweets en lien avec ces tendances

Fonctionnalité 3 : Organisation des données

## **Sprint 2 – Détection des insultes**

Fonctionnalité 4 : Récupérer et classer les listes d'insultes en anglais et en français

Fonctionnalité 5 : Détecteur d'insultes sur les tweets

Fonctionnalité 6 : Mise en forme des résultats

## **Sprint 3 – Présentation des résultats**

Fonctionnalité 7 : Notation des tweets

Fonctionnalité 8 : Affichage sur Dash des résultats

**MVP complété**

## **Sprint 4 – Amélioration détection insultes**

Fonctionnalité 9 : Adapter le code pour les tweets en anglais et trier par localisation

Fonctionnalité 10 : Pondérer la note en fonction de la "vulgarité" des tweets



# Répartition des tâches

- Fonctionnalité 1 : Collecte tendances
- Fonctionnalité 2 : Collecte tweets en lien avec ces tendances
- Fonctionnalité 3 : Organisation des données
- Fonctionnalité 4 : Récupérer et classer les listes d'insultes en anglais et en français
- Fonctionnalité 5 : Détecteur d'insultes sur les tweets
- Fonctionnalité 6 : Mise en forme des résultats
- Fonctionnalité 7 : Notation des tendances
- Fonctionnalité 8 : Affichage sur Dash des résultats
- Fonctionnalité 9 : Adapter le code pour différentes localisations
- Fonctionnalité 10 : Pondérer la note en fonction de la "vulgarité" des tweets
- Tout au long : développer des fonctions de test de nos fonctions

Légende :

Matthieu / Quentin / Étienne / Antoine / Lisa / Ambroise / Ensemble



# Structure du code

```
cw2_vulgarity
| --- Data
| --- test
| --- WorkingDocs
| --- src
| --- conception.md
| --- .gitignore
| --- README.md
| --- main.py
```

```
|--- test
| --- __init__.py
| --- test_api_connexion.py
| --- test_collect_trends.py
| --- test_collect_tweets.py
| --- test_tweet_data.py
```

```
| --- WorkingDocs
| --- tousantiinsultes.png
| --- Définition_du_travail.txt
```

```
| --- src
| --- Dash
| --- assets
| --- footer.css
| --- header.css
| --- tousantiinsultes.png
| --- typography.css
| --- __init__.py
| --- app.py
| --- tweet_analysis
| --- insultes_en.csv
| --- insultes_fr.csv
| --- rate_trends.py
| --- tweet_collect
| --- __init__.py
| --- api_connexion.py
| --- collect_trends.py
| --- collect_tweets.py
| --- json_file.py
| --- tweet_data.py
```





# Tests du code

```
----- coverage: platform win32, python 3.8.8-final-0 -----
Name                               Stmts  Miss  Cover
-----
main.py                             30     16    47%
src\Dash\__init__.py                 0      0   100%
src\Dash\app.py                      31     20    35%
src\tweet_analysis\rate_trends.py    35      1    97%
src\tweet_collect\__init__.py         0      0   100%
src\tweet_collect\api_connexion.py    8      0   100%
src\tweet_collect\collect_trends.py   9      0   100%
src\tweet_collect\collect_tweets.py   5      0   100%
src\tweet_collect\credentials.py      4      0   100%
src\tweet_collect\json_file.py       15      0   100%
src\tweet_collect\tweet_data.py      43      2    95%
test\__init__.py                     0      0   100%
test\test_api_connexion.py            3      0   100%
test\test_collect_trends.py           3      0   100%
test\test_collect_tweets.py           3      0   100%
test\test_tweet_data.py               5      0   100%
-----
TOTAL                               194     39    80%
```





# Doc

## L'équipe

- Lisa Lupi
- Antoine Dieu
- Etienne Lefranc
- Ambroise Marché
- Matthieu Scharffe
- Quentin Echasseriau

## Description du projet

Ce projet s'inscrit dans les projets de deuxième semaine des Coding Weeks de CentraleSupélec. Le programme récupère les tweets et les tendances twitter actuelles grâce aux fonctionnalités de l'API twitter et des modules pythons associés et renvoie un histogramme qui compare le taux d'insultes associé à chaque tendance. Ce taux est donné sous la forme du nombre d'insultes pour cent tweets, le tout pondéré selon la vulgarité et trié par type d'insulte. Cet outil peut être intéressant pour détecter les sujets les plus polémiques.



conception.md 1.62 KB



Éditer

Web IDE

Replace

Delete



Le but est de repérer les sujets en tendance qui engendrent le plus de tweets contenant des insultes et ainsi de détecter les sujets les plus polémiques.

## Sprint1 : Collecte et stockage de données.

- ☐ **Fonctionnalité1 :**  
collecter les tendances
- ☐ **Fonctionnalité2 :**  
collecter les tweets se rapportant à ces tendances et les sauvegarder dans un fichier .json
- ☐ **Fonctionnalité3 :**  
mettre ces données sous la forme d'un pandas dataframe