Rapport Projet The Office Classifier

Sommaire

Introduction

Ce rapport a pour but d'expliquer les différentes étapes du projets d'IA, les différentes réflexions et choix effectués, ainsi que les résultats obtenus.

J'ai choisi de travailler sur un dataset de dialogues de la série télévisée "The Office" comprenant 9 saisons.

Le but final de ce projet est de prédire le personnage qui parle en fonction de la réplique. On peut alors imaginer d'autres applications comme le fait de donner une phrase que l'on pourrait dire nous-même et de voir à quel personnage de la série elle correspondrait le plus.

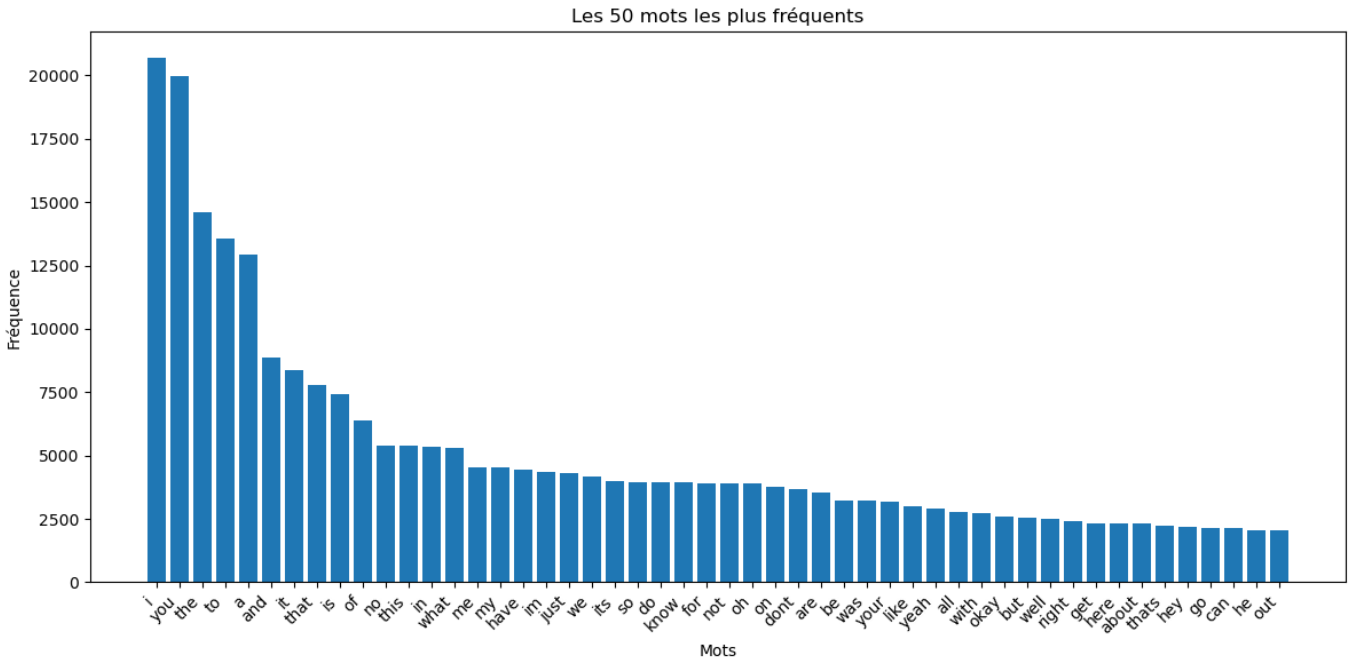
Visualisation des données

Avant de commencer à travailler sur les données, il est important de les visualiser pour comprendre leur structure et leur contenu.

J'ai donc choisi de générer 3 graphiques pour explorer les données du dataset

Comptage des mots

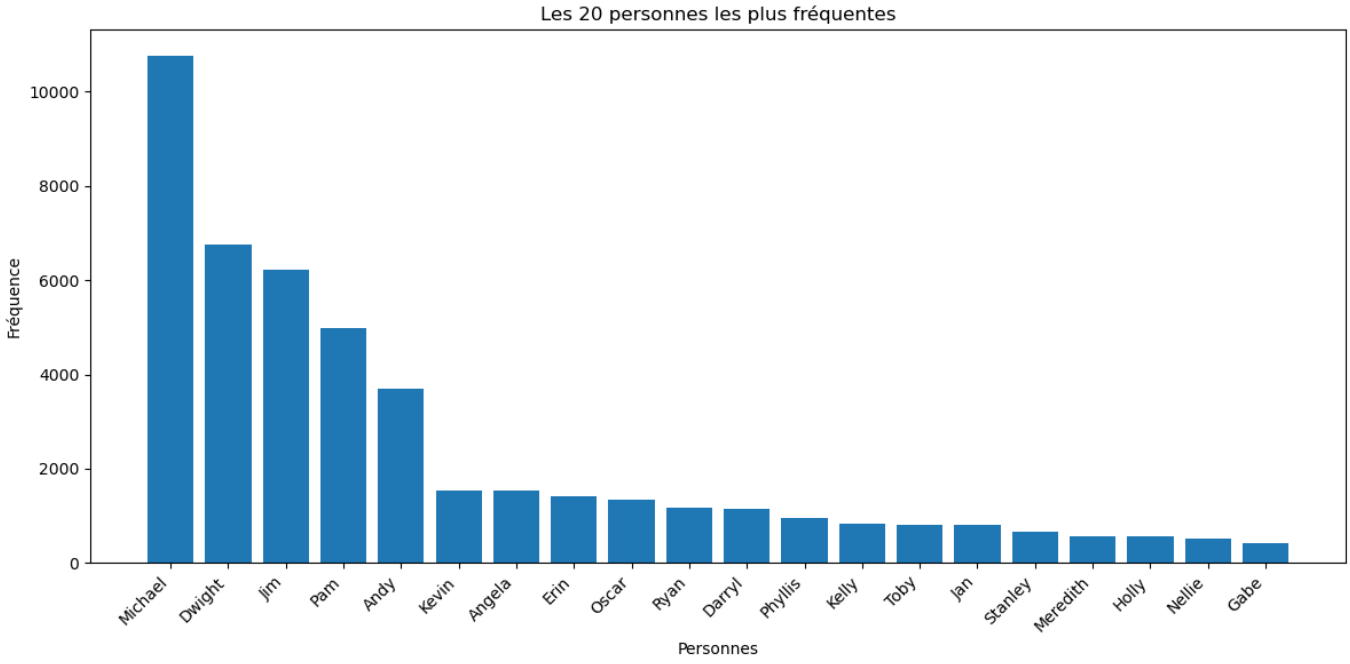
Le premier graphique représente le nombre total de mot sur l'ensemble des dialogues de la série.



Commme on pouvait s'en douter, les mots les plus fréquents sont des mots très courants que l'on appelle des stop words et qui n'apportent pas d'information. C'est pour cela que lors des pré-traitements on va les retirer pour ne pas fausser les résultats.

Personnages les plus fréquents

Ensuite, j'ai généré un graphique représentant les personnages les plus fréquents dans les dialogues.

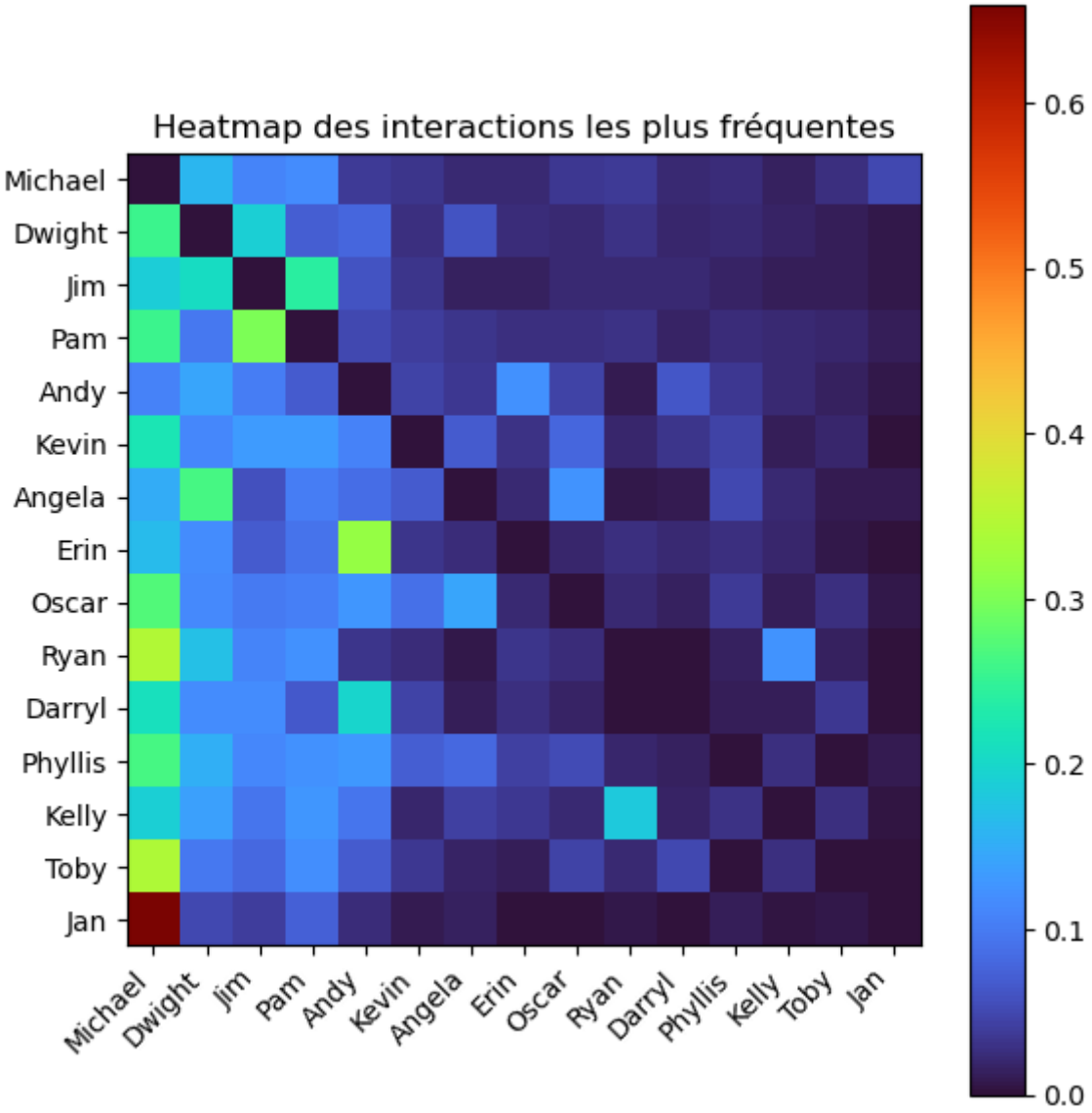


Si l'on connait la série, le graphique est assez logique, les 4 les plus présents dans les dialogues sont ceux que l'on voit toute la série et qui sont les principaux personnages.

Ensuite, les autres personnages sont plus ou moins présents en fonction de leur importance dans la série.

Interactions entre les personnages

Pour finir, une dernière visualisation des interactions entre les personnages.



On peut lire le graphique de la façon suivante : Michael est très présent dans les discussions de Jan, mais Jan n'est pas tant présente dans les discussions de Michael. (C'est dû au fait que Michael a beaucoup de dialogues avec de nombreux autres personnages donc Jan n'est pas si présente que cela dans les dialogues de Michael)

De même, on peut voir que Michael est présent pour beaucoup de personnages, ce qui est logique étant donné qu'il est le patron de l'entreprise.

Et pour finir, on peut observer une fréquence assez forte et réciproque entre Jim et Pam, ce qui est normal étant donné qu'ils sont très proches tout au long de la série

Pré-traitements

Pour pouvoir travailler sur les données, il est nécessaire de les pré-traiter pour les rendre exploitables par les modèles.

Nettoyage initial

Le nettoyage initial consiste à effectuer différentes étapes:

- Mettre en minuscule les mots
- Retirer les sauts de ligne
- Retirer les stopwords
- Retirer la ponctuation.

De plus, on ajoute une étape de lemmatization pour ne garder que la racine des mots à l'aide du dictionnaire `en_core_web_sm` de spacy.

Paramètres de data préparation

Au-delà du nettoyage initial, j'ai identifié, notamment grâce aux visualisations, 2 paramètres sur lesquels jouer pour améliorer les résultats des modèles.

1.Le premier est le `nombre de dialogues par personnage`. En effet comme nous l'avons vu avec la deuxième visualisation, certains personnages ont beaucoup plus de dialogues que d'autres, ce qui peut fausser les résultats des modèles. De plus, certains personnages n'ont pas assez de dialogues pour être représentatifs et entrainer les modèles. J'ai donc décider de limiter dialogues par personnage. Nous testerons alors trois valeurs différentes :

- 500 dialogues
- 750 dialogues
- 1000 dialogues

2.Le deuxième paramètre est le `nombre de caractères par dialogue`. En effet, certains dialogues sont très courts et ne contiennent pas assez d'information pour entrainer les modèles. J'ai donc décidé de limiter les dialogues à différents nombres de caractères :

- Entre 35 et 75
- Entre 45 et 85
- Entre 25 et 85
- Au dessus de 40
- Au dessus de 55
- Entre 20 et 55

Vectorisation et division des données

Enfin, pour chaque combinaison de paramètres, j'ai vectorisé les données à l'aide de la méthode `TfidfVectorizer` de scikit-learn et divisé les données en un jeu d'entrainement et un jeu de test.

Modèles

Pour entrainer les modèles, j'ai choisi d'utiliser 4 modèles différents :

- SVM
- Logistic Regression
- Random Forest
- Gradient Boosting

J'ai également expérimenté d'autres modèles allant jusqu'au Réseau de Neurones, mais les résultats n'étaient pas concluants par rapport au temps d'entrainement et vis a vis de la complexité de compréhension et d'analyse que cela apportait.

J'ai donc conservé ces 4 modèles qui sont des modèles classiques et qui me permettront d'avoir donc des résultats plus fiables.

Comparaison des combinaison des pré-traitements possibles

Je pouvais alors combiner les différents paramètres de pré-traitements que j'ai entrainés sur chaque modèle.

J'ai ensuite fait la moyenne de l'accuracy des modèles pour avoir une idée des combinaisons les plus efficaces.

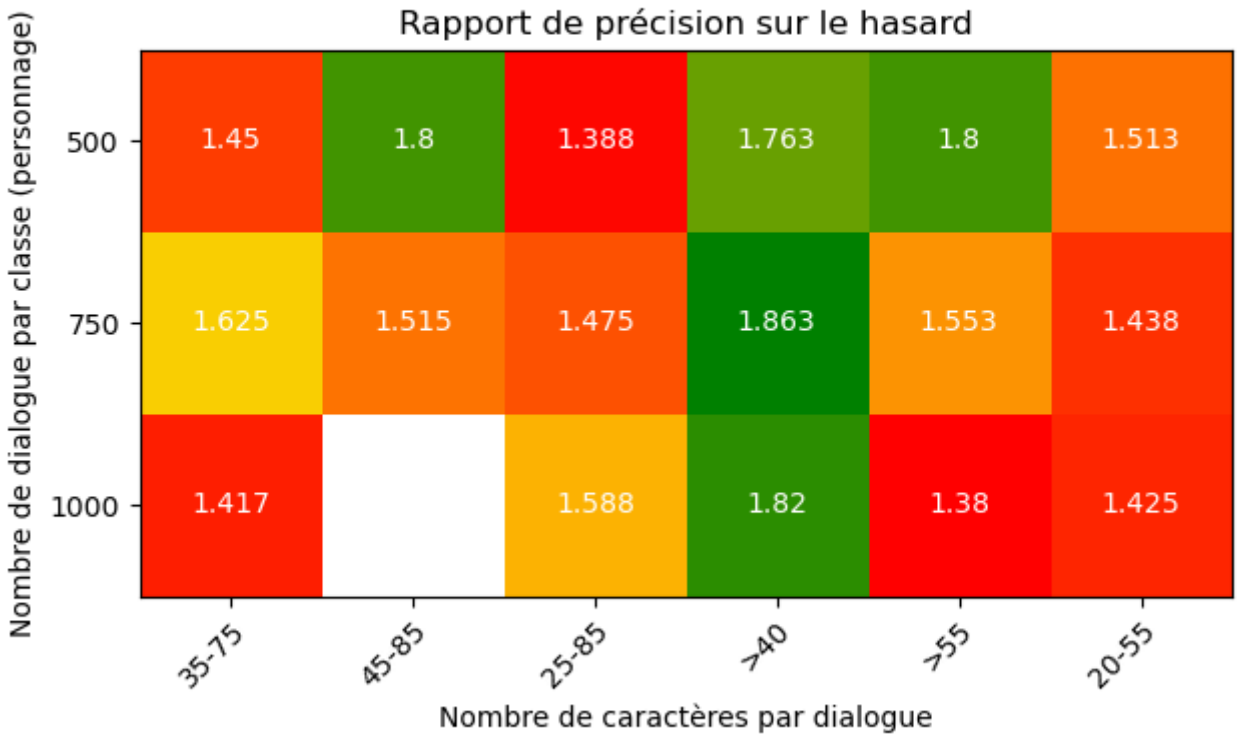
Ensuite, j'ai calculé à partir de cette moyenne ce que j'appelerais un `rapport de précision sur le hasard`, qui me permettrait de déterminer combien de fois le modèle est plus précis que s'il avait prédit au hasard.

Par exemple, si l'accuracy d'un modèle est de 0,37 et que le nombre de classe est de 5, alors ce ratio est de 0,37 / 0,2 = 1,85

Cela signifie que le modèle est 1,85 fois plus précis que s'il avait prédit au hasard.

Donc plus ce ratio est élevé, plus le modèle et sa combinaison de paramètres de data préparation est précis.

Voici les résultats obtenus :



Concernant la case blanche combinant 1000 dialgues par personnage et un nombre de caractères entre 45 et 85, c'est tout a fait normal car cette combinaison ne résultait en un dataset contenant une seule classe.

Par ailleurs, en fonction de la combinaisons, un certain nombres de classes étaient présentes dans le dataset, allant de 5 à 3.

En analysant les résultats, on peut voir clairement que le meilleur paramètre pour le nombre de caractères par dialogue est `au dessus de 40`.

Concernant le paramètre du nombre de dialogue par personnage, il n'y a pas de conclusion claire, mais on peut voir que `500 dialogues` est souvent le meilleur paramètre.

Pour notre cas, la meilleure combinaison est finalement de `750 dialogues` par personnage et `au dessus de 40` caractères par dialogue.

En regardant les résultats de chaque modèle sur cette combinaison de paramètres, on a les accuracy suivantes :

- 0.39 : SVM
- 0.40 : Régression Logistique
- 0.35 : Random Forest
- 0.35 : Gradient Boosting

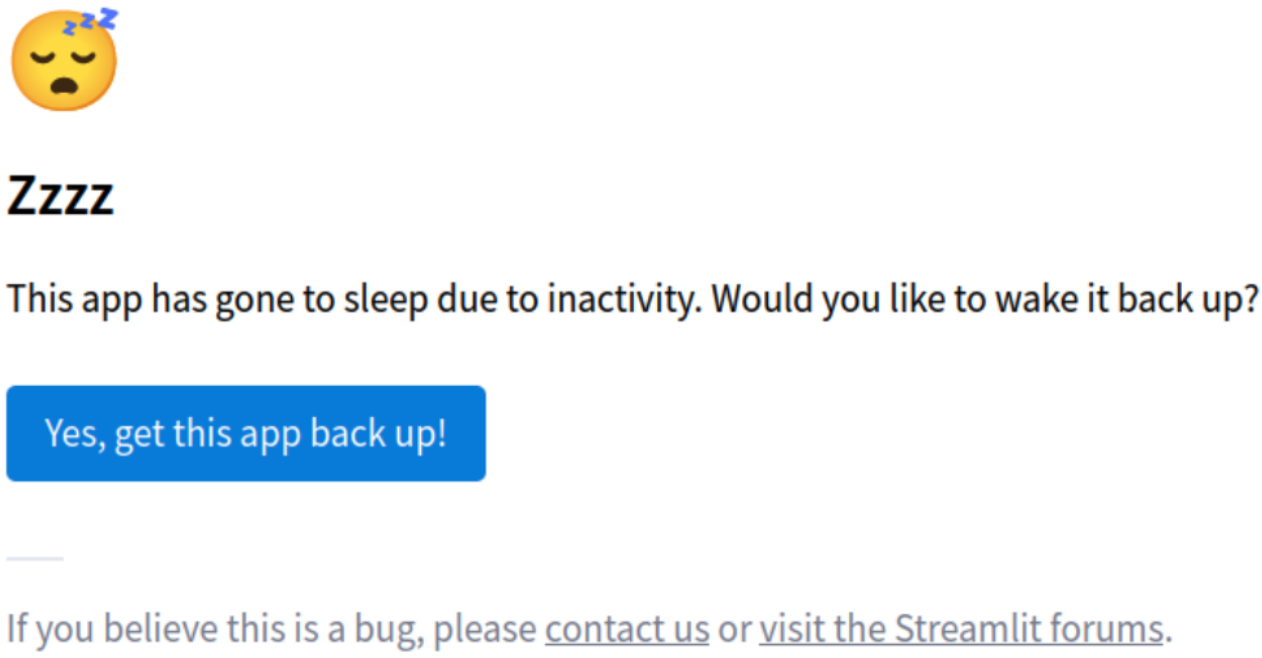
On prendra donc la Régression Logistique pour notre modèle final.

Bonus: Application Streamlit

En plus de cela, j'ai pu créer une application Streamlit qui permet de prédire le personnage qui parle en fonction de la phrase que l'on rentre (en Anglais).

L'application est disponible sur le lien suivant : Lien vers l'application

⚠ Il est fort probable que l'application soit éteinte lorsque vous cliquerez sur le lien car elle est hébergée sur le serveur gratuit de Streamlit qui s'éteint après un certain temps d'inactivité. Si c'est le cas, il suffit alors simplement de cliquer sur le bouton présent sur le lien pour la relancer :



Si besoin, le code pour l'application est disponible sur le repo GitHub suivant : Lien vers le repo GitHub