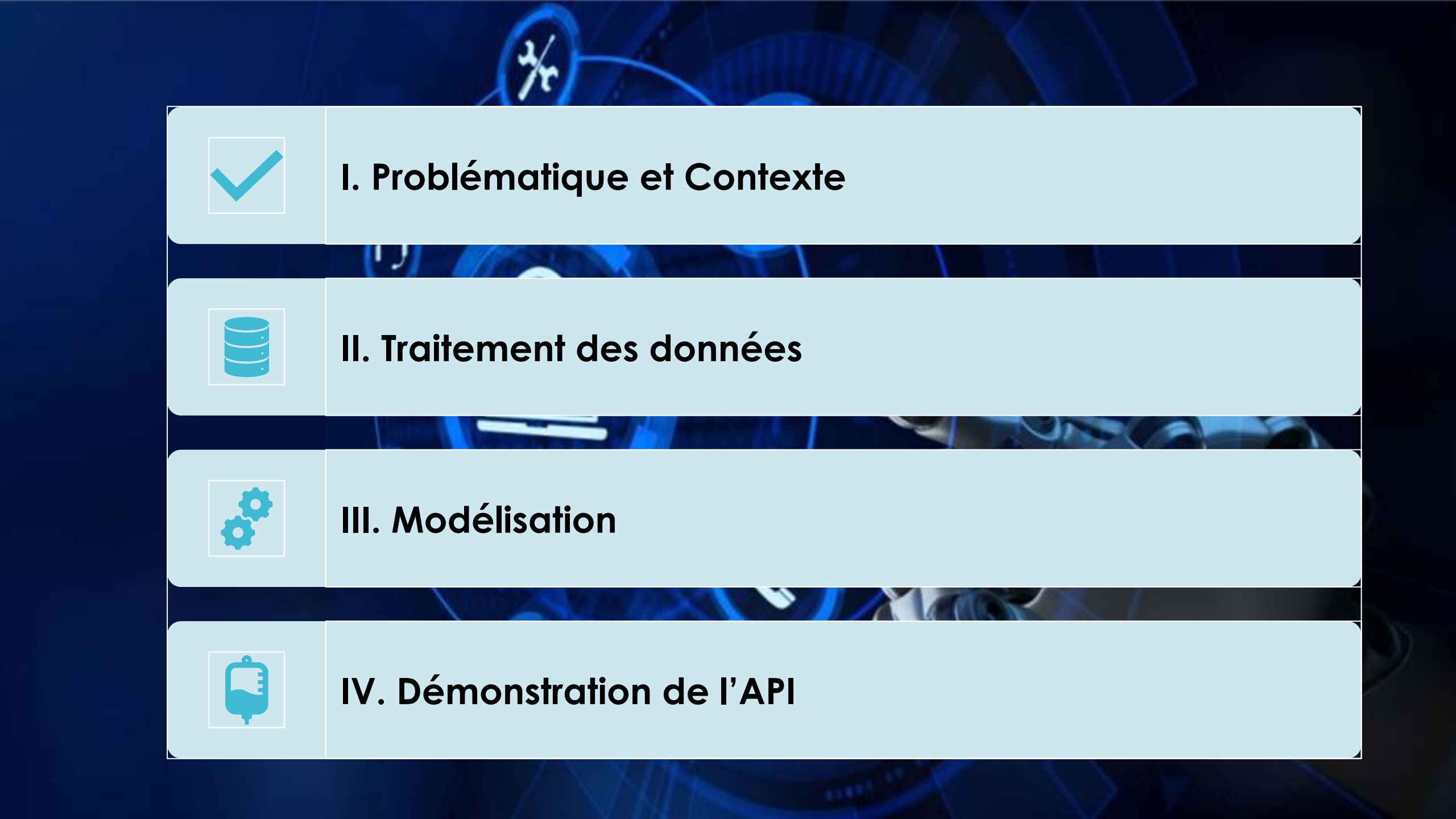
Projet 5 :
Catégorisez automatiquement des questions

OPENCLASSROOMS

Cédric Randrianarivélo



I. Problématique et Contexte

II. Traitement des données

III. Modélisation

IV. Démonstration de l'API

I. Problématique et Jeu de données

OPENCLASSROOMS

Contexte

- Stack Overflow est une plateforme célèbre de **questions-réponses** qui permet aux utilisateurs de partager leurs connaissances et d'aider la communauté.
- Le but de ce projet est d'aider la plateforme à **générer des tags pertinents** pour les questions posées, ce qui améliorera l'expérience utilisateur en proposant des suggestions de tags précises.
- Nous allons utiliser **les questions posées** sur la plateforme pour proposer des tags adaptés, en nous appuyant sur les données d'exportations disponibles via l'outil "StackExchange Data Explorer". Ces données authentiques fourniront un grand nombre d'informations pour **améliorer la pertinence** des tags proposés.
- L'objectif de ce projet est de développer un modèle de NLP (traitement automatique du langage naturel) capable de prendre en entrée des questions et un corps de texte, et de renvoyer des tags pertinents **via une API**. Cela permettra aux utilisateurs de Stack Overflow de bénéficier de suggestions de tags précises et adaptées à leur question, améliorant ainsi l'expérience utilisateur.

II. Traitement des données

OPENCLASSROOMS

Etapes de traitement du jeu de données.

Traitement de texte

- Conversion en minuscule
- Retrait de la ponctuation des mots

Traitement format HTML

- Retrait des caractères HTML du texte
- Autre fonction pour retirer le format HTML sur le Tags

Filtrage sur la longueur des mots

- Retire les mots du textes inférieurs à 2 caractères
- Retire les mots du textes supérieurs à 21 caractères.

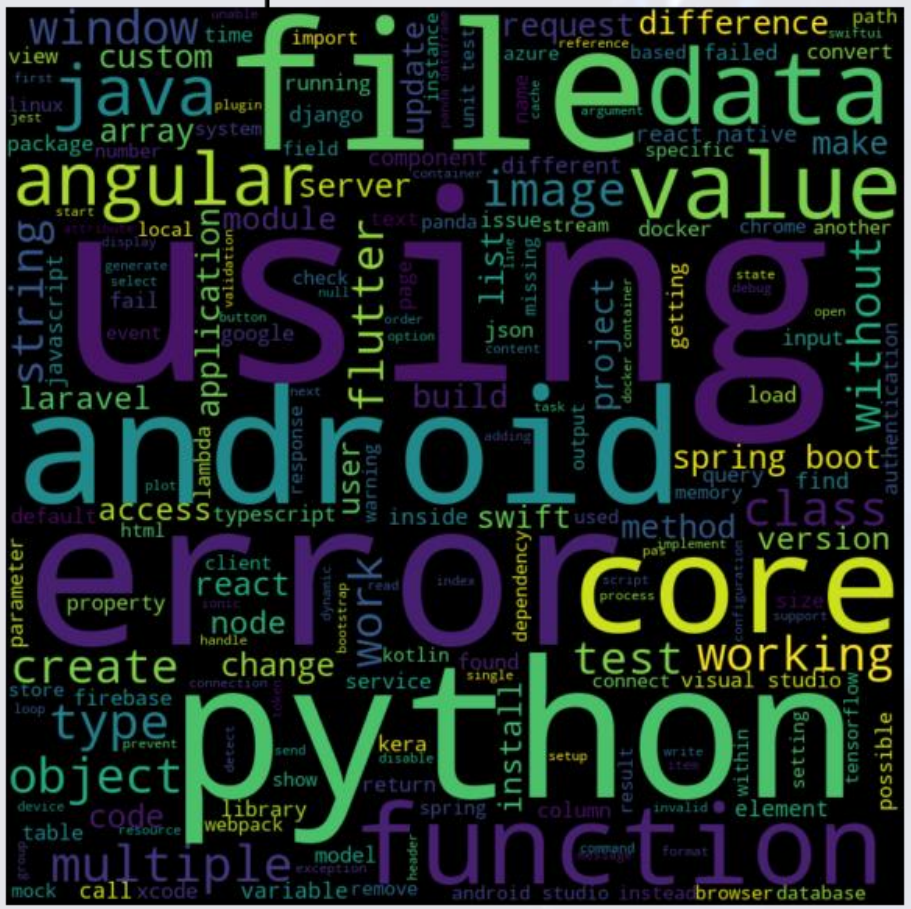
Tokenization du texte

- Découpage des textes en format « tokens »
- Retire les mots du texte contenu dans le « Stop Words »

Lemmatization du textes

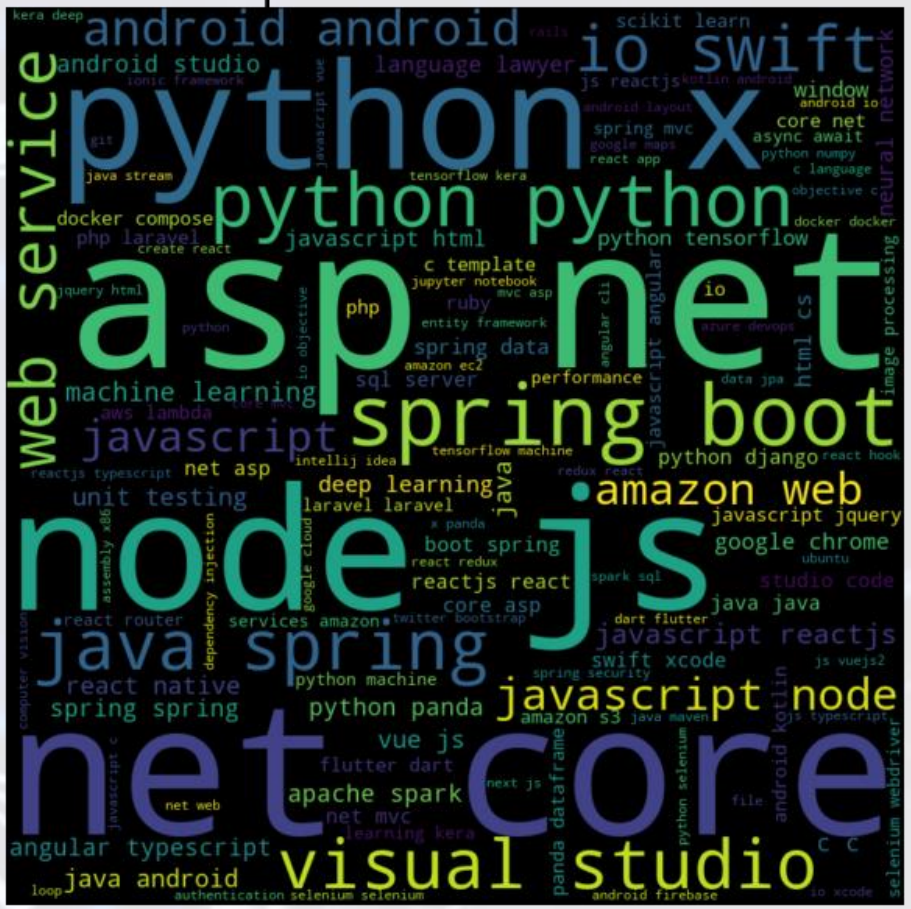
- Récupère les mots sans accords et conjugaisons. Forme radicale du mot

Top words in reviews

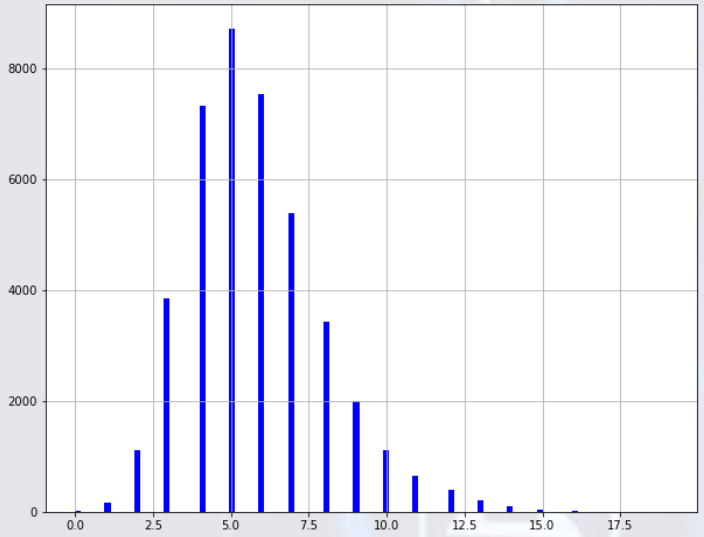


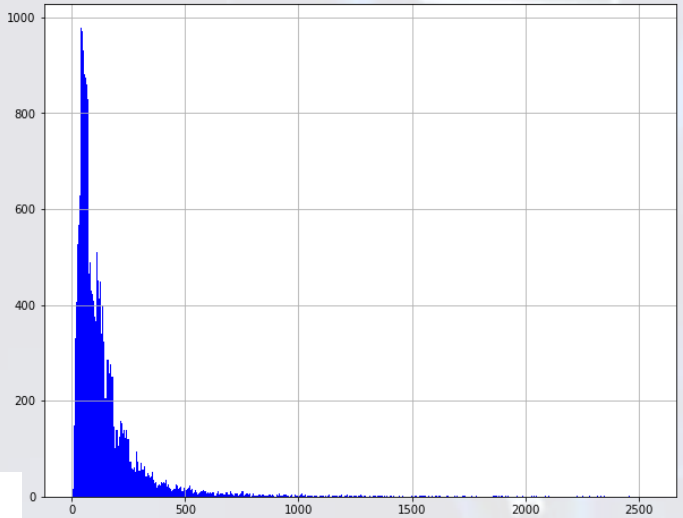
Mots les plus fréquents dans les titres des questions.

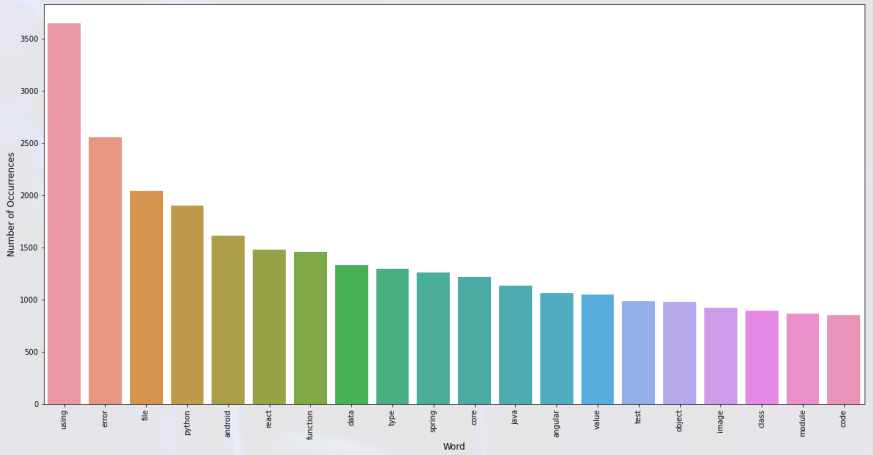
Top words in reviews



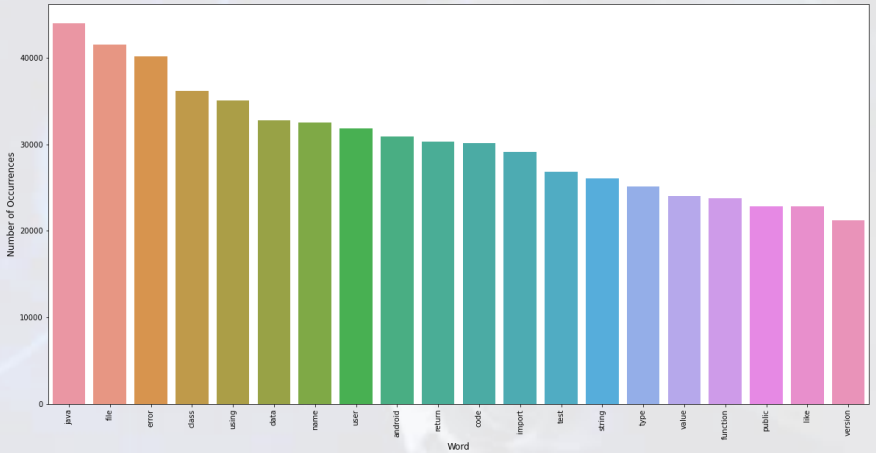
Les tags les plus fréquents.







Titre des questions



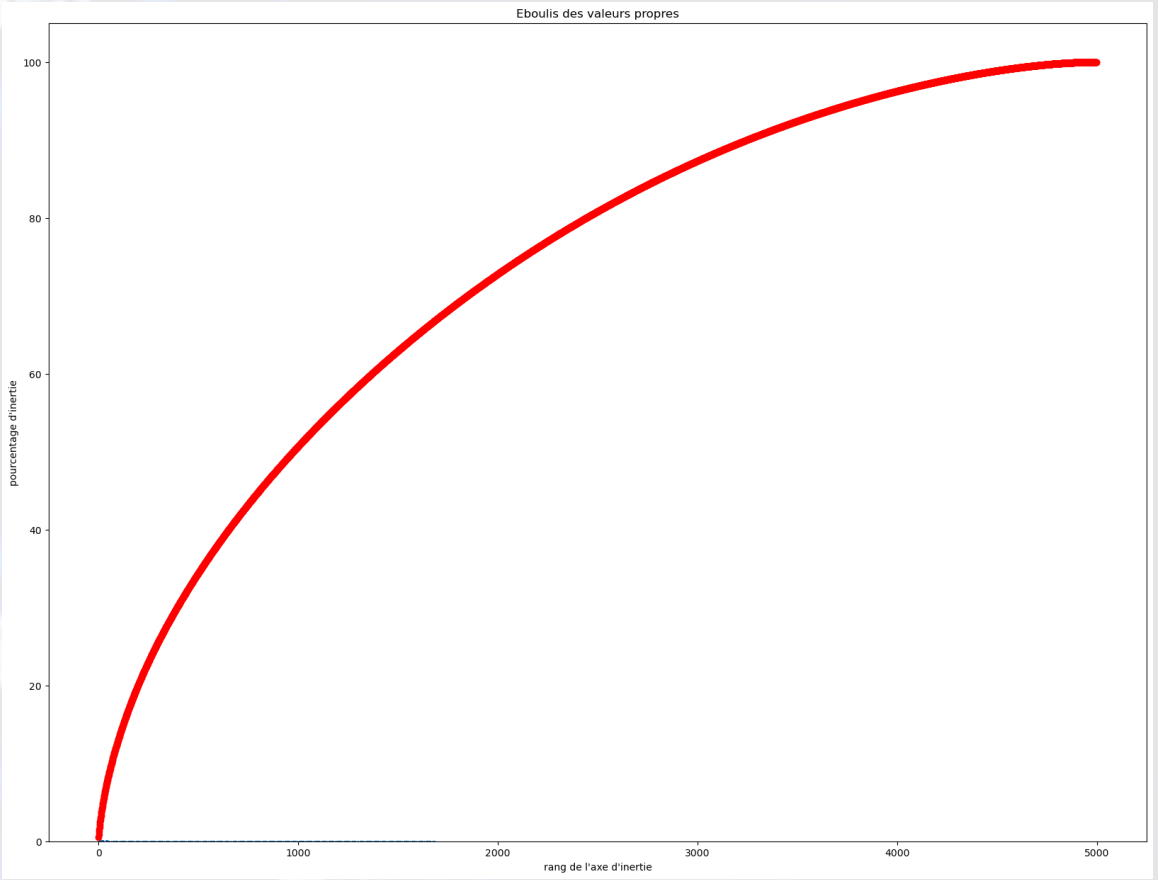
Corps des questions

	Value
python	1976.675877
android	1662.825082
java	1273.777683
javascript	1265.054985
net	1155.229268

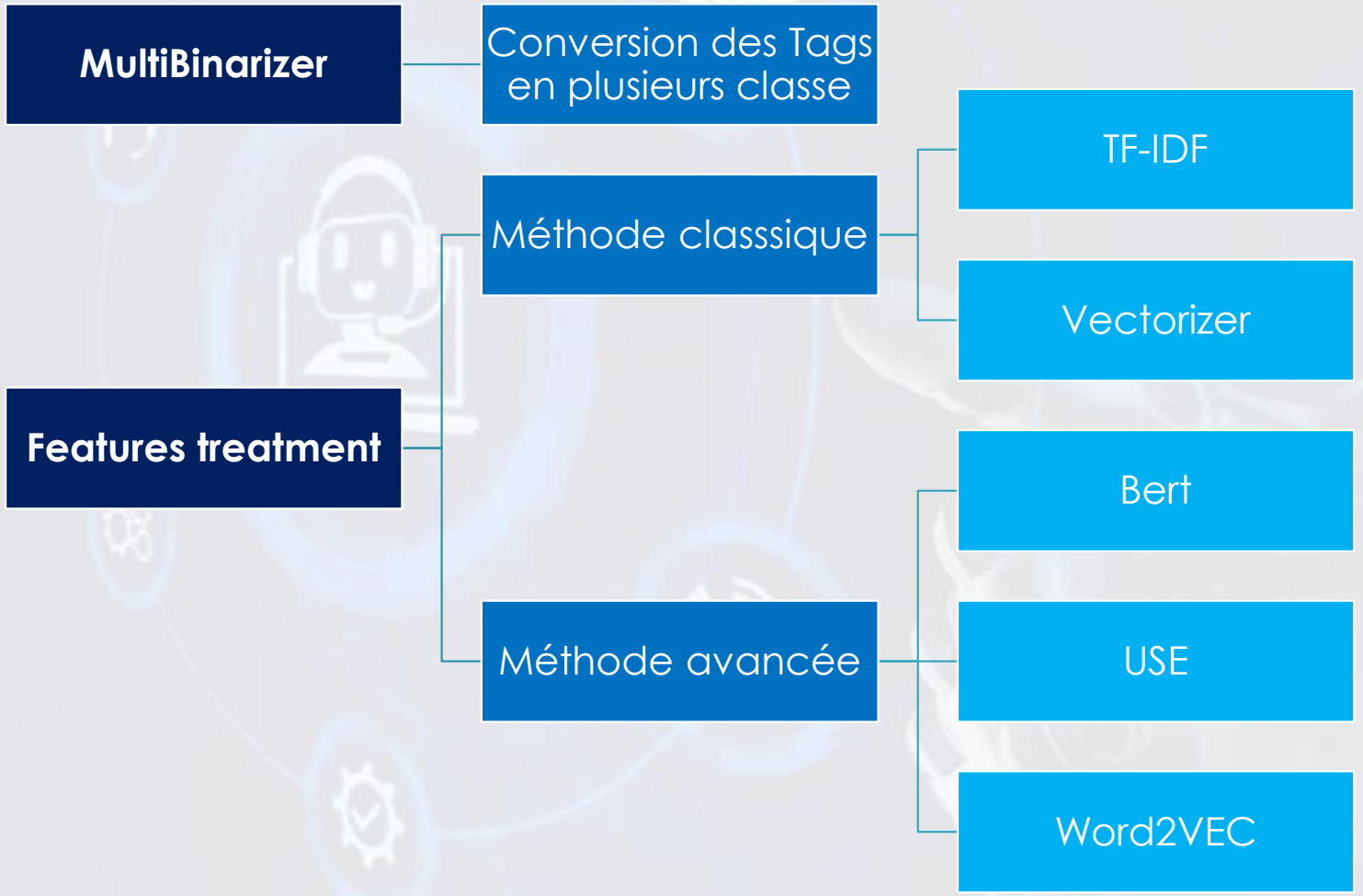
III. Modélisation

OPENCLASSROOMS

Choix du nombre de features



Transformation des features



Approches utilisées

Approche non-supervisée

Coefficient de cohérence

Modèle

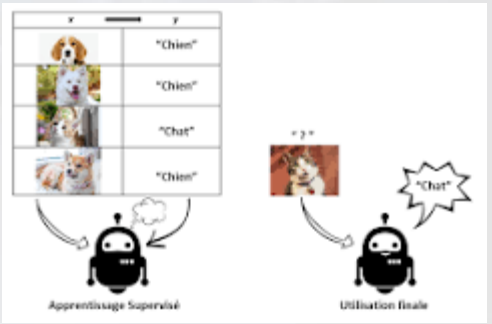
- *NMF*
- *LatentDirichlet Allocation*

Approche supervisée :

Jaccard Score

Modèle

- *SGD Classifier*
- *Passsive Agresssive Classifier*
- *Logistic Regression*

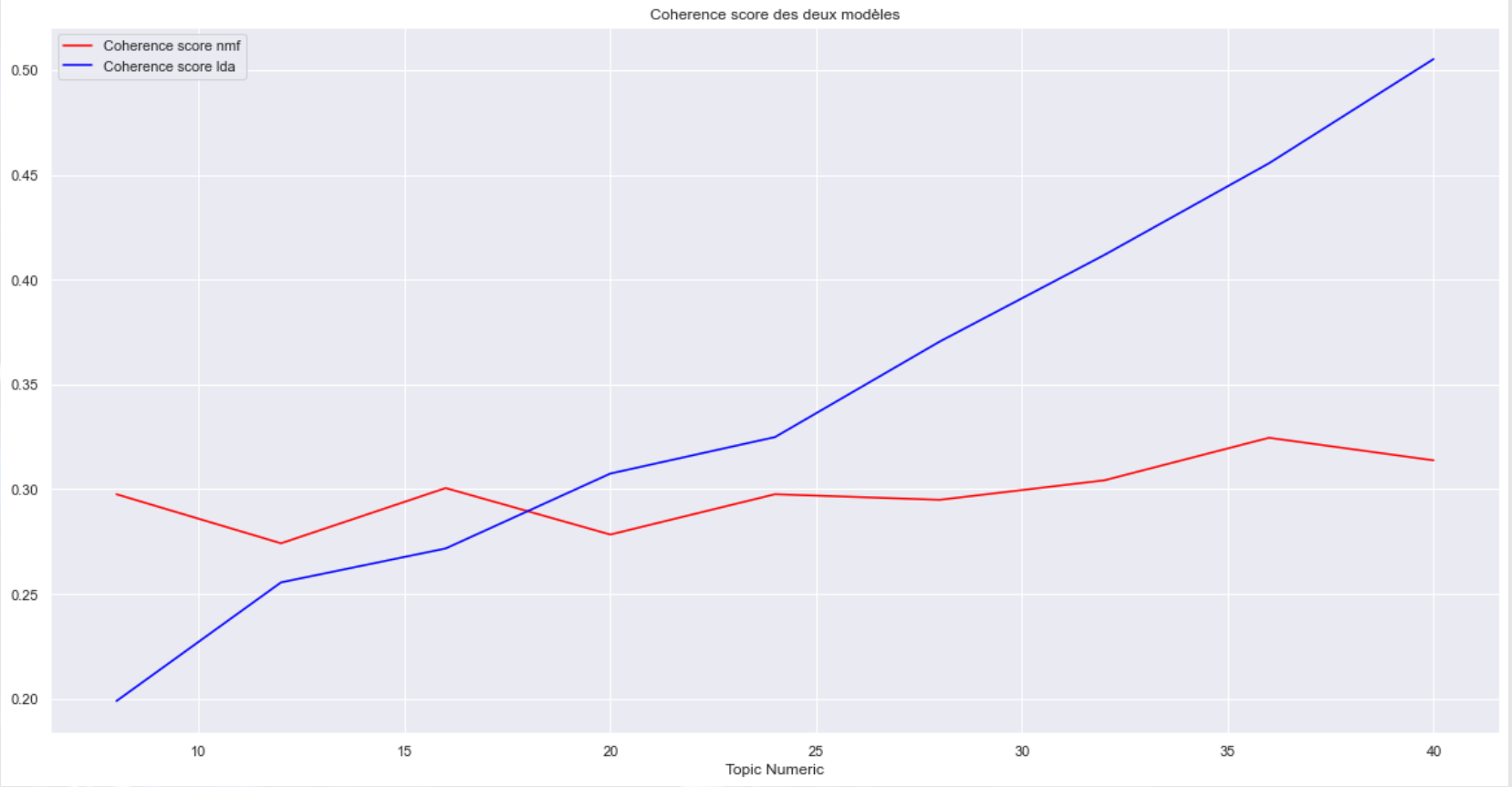


Approche non-supervisée

- Tracer du coefficient de cohérence
- en fonction du nombre de topic

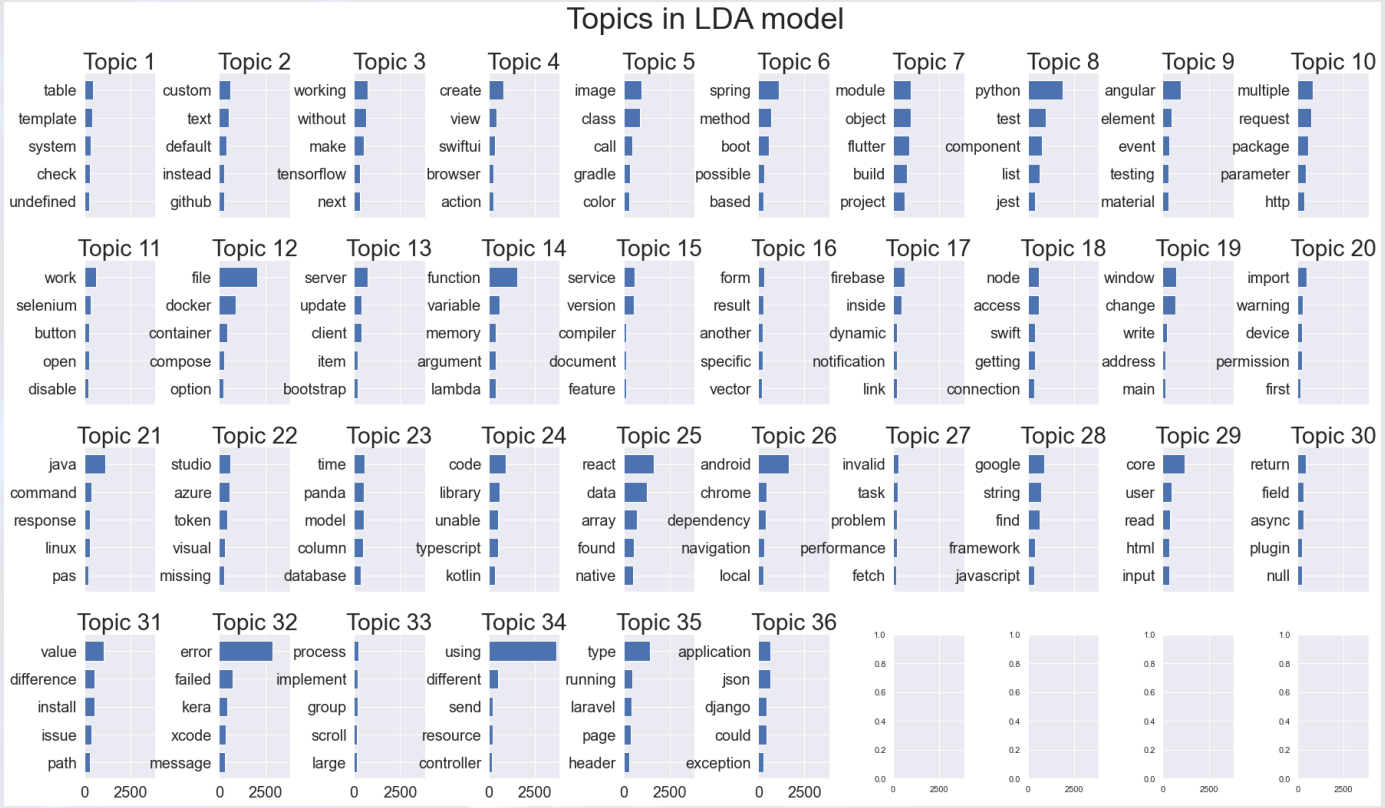
- Choix du meilleur modèle

```
NMF model Best topic:40
LDA model Best topic:36
```



Approche non-supervisée

- **Choix modèle** :
  - Modèle choisi : LDA
  - Moins de Topics
  - Avec le score idéal de cohérence
- **Performance** :
  - Jaccard score : 0.02
  - Calcul fait sur les tags
  - Score peu pertinent
  - Se tourner vers les modèles supervisés




```
{'Jaccard_score': [0.025437967721933512],
 'accurracy': [0.0],
 'recall': [0.025605886524454752],
 'precision': [0.7950411379857256],
 'f1': [0.04961385968269802],
```

Approche supervisée

- Choix du nombre de Tags.
- Evaluation du « Jaccard Score » en fonction des targets.
- Plus il y a de Tags, plus le score baisse.
- Partir sur 50 Tags pour avoir une bonne précision.

Approche supervisée

- Paramètres choisis :
  - Choix du Modèle SGD
  - Choix du feature treatment : TF_IDF

	Model	Features Traitement	Jaccard	accuracy	recall	precision	f1	fit time
0	Model Logistic Regression	TF IDF	0.450950	0.509459	0.500000	0.821328	0.621593	83.330092
0	Model Logistic Regression	Vectorizer	0.433632	0.448191	0.565561	0.650217	0.604942	224.392075
0	Model SGD	TF IDF	0.488060	0.533682	0.541216	0.832476	0.655968	34.152372
0	Model SGD	Vectorizer	0.420233	0.414153	0.603218	0.580769	0.591780	152.779223
0	Model PAC	TF IDF	0.453161	0.433151	0.637386	0.610569	0.623690	107.161999
0	Model PAC	Vectorizer	0.418415	0.403942	0.614607	0.567243	0.589976	129.307243
0	Model Logistic Regression	Use	0.433632	0.448191	0.565561	0.650217	0.604942	227.778593
0	Model SGD	Use	0.420233	0.414153	0.603218	0.580769	0.591780	180.013873
0	Model PAC	Use	0.418415	0.403942	0.614607	0.567243	0.589976	128.994937
0	Model Logistic Regression	Use	0.460911	0.511517	0.522921	0.795366	0.630991	18.932030
0	Model SGD	Use	0.460442	0.515792	0.519505	0.801978	0.630551	7.978070
0	Model PAC	Use	0.422058	0.437584	0.550256	0.644328	0.593588	15.420245
0	Model Logistic Regression	Word 2Wec	0.340674	0.426898	0.398633	0.700876	0.508213	24.211955
0	Model SGD	Word 2Wec	0.331176	0.399272	0.411589	0.628957	0.497569	24.281457
0	Model PAC	Word 2Wec	0.272980	0.295575	0.419206	0.439019	0.428884	8.521948
0	Model Logistic Regression	Bert	0.180461	0.350000	0.192475	0.743017	0.305747	4.126655
0	Model SGD	Bert	0.217656	0.165000	0.413893	0.314631	0.357500	2.129656
0	Model PAC	Bert	0.235401	0.235000	0.373372	0.389140	0.381093	2.510931

IV. Démonstration de l'API

OPENCLASSROOMS

Démonstration

Logiciel : Utilisation de « Streamlit » et « Github »

Lien : https://cedricrandrianarivelo-cat--randrianarivelo-cedric-api-p5-z8ydce.streamlit.app/

Démonstration

Question :
More details about the nature of the problem and specific error messages would be needed to help diagnose the underlying cause of the error.

Corpus :
More details about the nature of the problem and specific error messages would be needed to help diagnose the underlying cause of the error.

It is also possible that some Python libraries are not compatible with the version of Android being used, which could cause problems. In general, the Python development community is very active and there are many resources online for help with troubleshooting Python programming issues on Android.

The user has a problem with his Python script when he deploys it to the cloud.He uses a Windows PC and an Android Google device.When he uses his Android device with his programming scripts, he consistently gets errors when he wants to implement functions via IO.He asks for help to solve this problem.To better understand the nature of the error, it would be helpful to know the specific error messages the user is getting.It is also important to know what cloud platform the user is using to host their Python application.It is possible that some IO functions are not supported on the user's Android Google device, which could cause errors.The code should be written to handle potential errors and provide clear error messages to help diagnose problems.It is recommended to use debugging tools such as message printers to help identify errors in the code.In general, the Python development community is very active and there are many online resources for help with troubleshooting on Google.

Merci votre attention !

Question ?

OPENCLASSROOMS