CATÉGORISEZ AUTOMATIQUEMENT DES QUESTIONS

BOURBON VICENTE

PROBLÉMATIQUE

- Stack Overflow: site de questions-réponses liés au développement informatique
- Développer un système de suggestions de tags
- Problème de classification multilabels
- Base de données de questions à disposition

SOMMAIRE

- Nettoyage et Feature engineering
- Exploration des données
- Modélisation non supervisée et semi-supervisée
- Modélisation supervisée
  - Avec représentation bag of words
  - Avec représentation sentence embedding
- API de prédiction des tags
- Conclusion

Contoso

I. NETTOYAGE ET FEATURE ENGINEERING

- Requête à partir de la base de données StackOverflow
- Sélection de questions pertinentes
- Ni valeur manquante, ni doublon, ni outlier

```sql
SELECT Title, Body, Tags, Id, Score, ViewCount, FavoriteCount, AnswerCount
FROM Posts
WHERE PostTypeId = 1 AND ViewCount > 10 AND FavoriteCount > 0
AND Score > 5 AND AnswerCount > 0 AND LEN(Tags) - LEN(REPLACE(Tags, '<','')) >= 5
```

FEATURE ENGINEERING

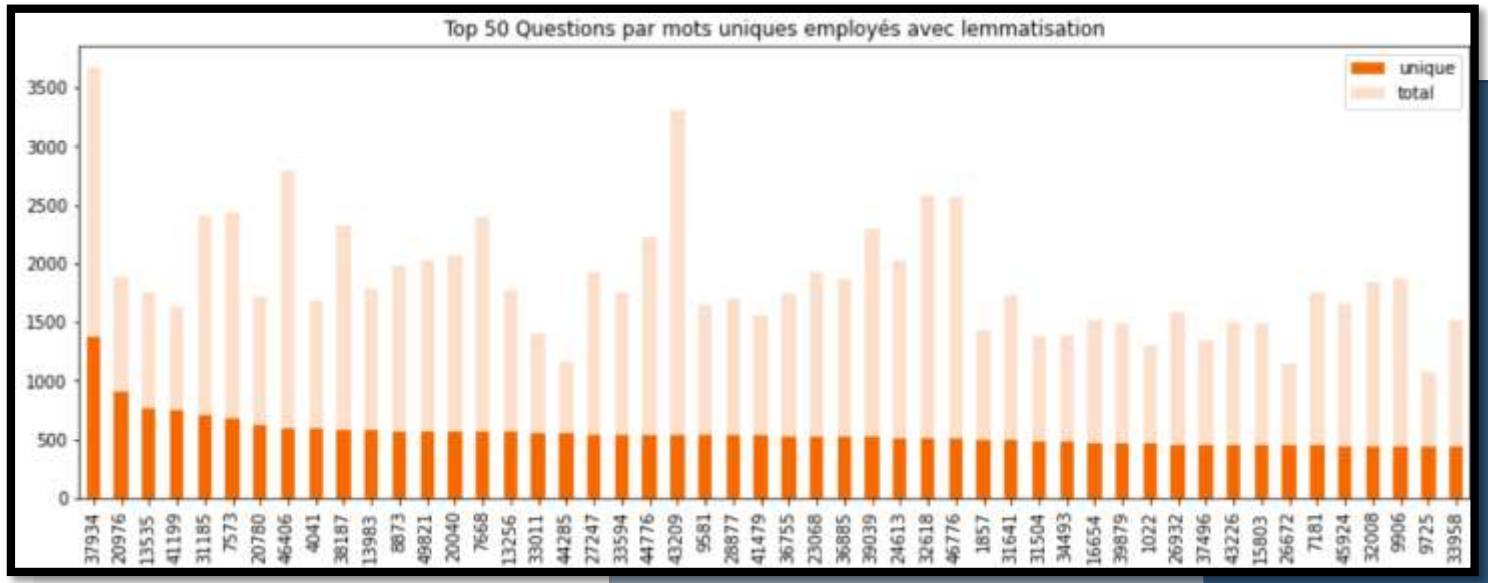
- Nettoyage HTML
- Formatage des tags
- Tokennisation
- Suppression des majuscules
- Suppression stop words
- Normalisation
- Séparation jeux entraînement/test pour les titres et les questions

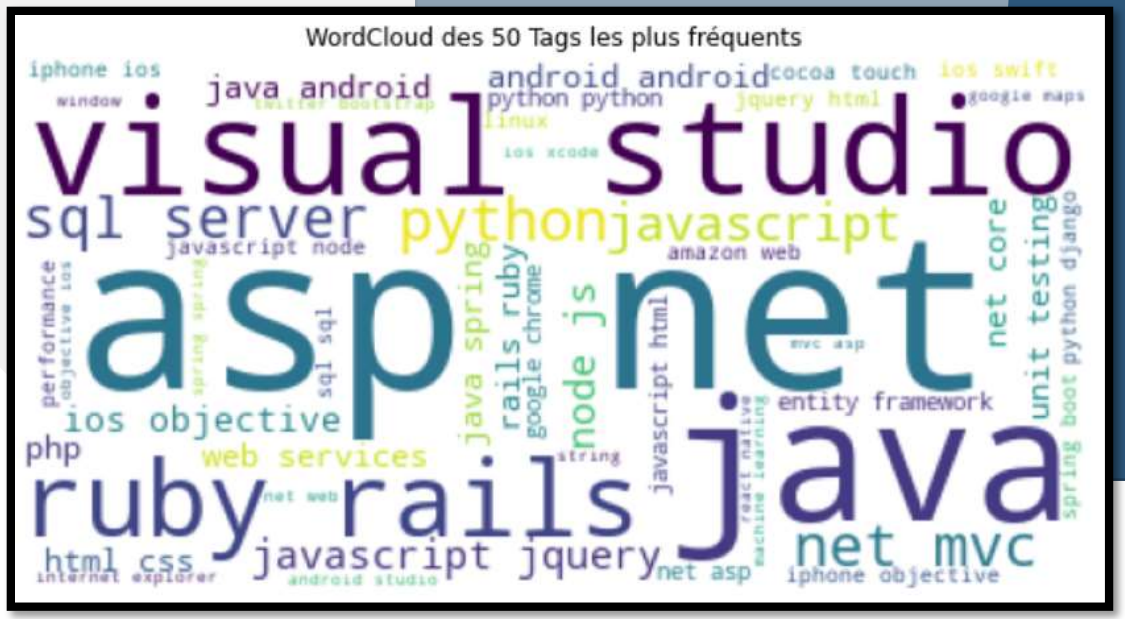
```
'We have a C web app where users'
['We', 'have', 'a', 'C', 'web', 'app', 'where', 'users']
['we', 'have', 'a', 'c', 'web', 'app', 'where', 'users']
['c', 'web', 'app', 'users']
['c', 'web', 'app', 'user']
```

Contoso

II. EXPLORATION DES DONNÉES

- Nombre et fréquence des mots dans le titre et la question en fonction des méthodes de nettoyages
- Nombre de Tags par question
- Fréquence des Tags
- Corrélation des variables Nombre de vues et score

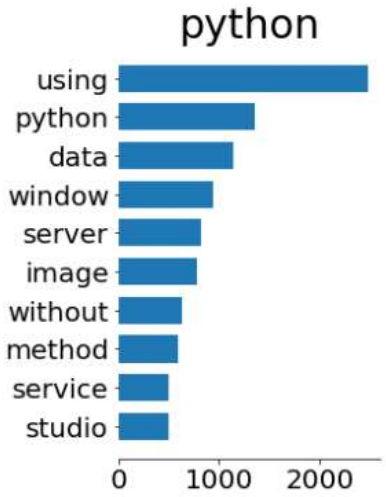


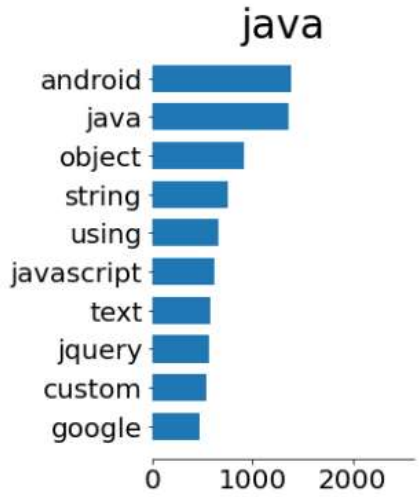


III. MODÉLISATION NON SUPERVISÉE ET SEMI-SUPERVISÉE

NON SUPERVISÉE: LDA

- Détection automatique de sujets
- GridSearch avec 5, 10 et 15 topics
- Association du tag le plus fréquent pour les questions/titres du topic
- Proportion de questions/titres par topic





SEMI-SUPERVISÉE

- Association d'un topic à chaque question/titre du jeu test
- Tag prédit: tag le plus fréquent du topic
- Contrôle si le tag prédit appartient à la liste des tags réels

EXEMPLE

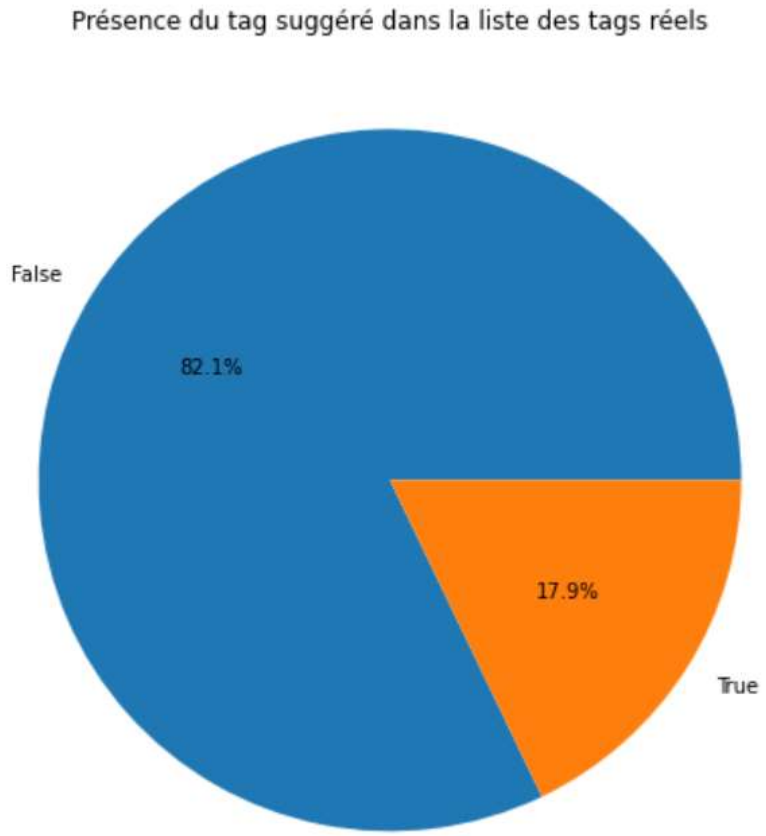
- Titre: How ADD Schema VideoObject in Wordpress if iframe Youtube Exist
- Tags: php, wordpress

- Topic prédit: second topic
- Tag prédit: Java
- Java non présent dans les tags réels

RÉSULTATS

CRITIQUES DU MODÈLES

- Tags non caractéristiques d’un topic
- Trop de sujets différents
- Nombre de topics élevé inexploitable en pratique
- Intéressant pour de l’exploration mais pas pour de la prédiction



PROPORTION DE BONNE PRÉDICTION

IV. MODÉLISATION SUPERVISÉE

1. AVEC REPRÉSENTATION BAG OF WORDS

- Utilisation des 100 mots/tags les plus fréquents du corpus
- Entraînement d'un classifieur pour chaque tag
- Probabilité d'appartenance à chaque classe(=tag) pour les questions/titres du jeu de test
- Comparaison des modèles via temps de calcul et score roc auc

Modèles

- Direct Matching
- Naïve Bayes
- Arbre de Décision
- Forêt Aléatoire
- Régression Logistique

Contoso

EXEMPLE DE DIRECT MATCHING

- Titre: How to segment text from a .txt file using Spacy in Python?
- Tags prédits: file, python

RÉSULTATS

AVEC LES TITRES

stats_titles

	temps calcul	roc auc
direct matching	0.220281	0.588864
naive bayes	189.204985	0.829291
decision tree	145.931799	0.764966
random forest	11020.770731	0.80589
logistic regression	134.458623	0.837063

- Direct Matching intéressant à utiliser en complément d'un autre modèle
- Choix de la régression logistique avec les questions

AVEC LES QUESTIONS

stats_questions

	temps calcul	roc auc
direct matching	0.082623	0.572171
naive bayes	173.253978	0.840685
decision tree	139.677479	0.587695
random forest	11264.935529	0.846012
logistic regression	134.337904	0.866082

Contoso

IV. MODÉLISATION SUPERVISÉE

1. AVEC REPRÉSENTATION SENTENCE EMBEDDING

- Méthodes de réduction dimensionnelle
- Prise en compte du sens des mots et des phrases
- Utilisation de 3 méthodes d'embedding: Doc2Vec, Bert, USE
- Entraînement d'un classifieur pour chaque tag
- Comparaison des modèles via temps de calcul et score roc auc

Modèle

- Régression Logistique
- Avec les questions

RÉSULTATS

stats_embeddings

	temps calcul	roc auc
Doc2Vec	187.506971	0.906374
Bert	213.061147	0.94757
USE	200.078237	0.964791

Meilleurs scores roc mais du sur-apprentissage

Temps de calcul raisonnables

Sélection du modèle final de régression logistique avec représentation bag of words des questions

Contoso

V. API DE PRÉDICTION DES TAGS

render

UTILISATION

FONCTIONNEMENT

- Saisie du titre et de la question
- Méthode de direct matching sur le titre et la question
- Régression logistique sur la question
- Proposition de tags

[FORMULAIRE STACKOVERFLOW (P5APPBOURBON.ONRENDER.COM)]

CONCLUSION

- Enseignements:
  - Travail sur des données textuelles: tokenization, normalisation, représentation
  - Principes de la classification multilabels
  - Déployer une application
- Problèmes rencontrés:
  - Appréhender et mettre en place la classification multilabels
  - Déployer une application
- Pistes d'amélioration:
  - Utilisation de OneVsRestClassifier de sklearn
  - Optimisation des hyperparamètres
  - Périmètre de travail (nombre de tags/mots)

Contoso