OpenClassrooms

Soutenance: Projet 5

Stacks Overflow - Tagging Supervise & Non-Supervise

Edward Levavasseur

Updated: 2021/05/12

Sujet

Problématique Stacks Overflow

Amateur de Stack Overflow [...] vous développez un système de suggestion de tag pour le site. Celui-ci prendra la forme d'un algorithme de machine learning qui assigne automatiquement plusieurs tags pertinents à une question.

- Approche Supervisée
- Approche Non-Supervisée

Plan

1. Création d'un Bag of Words
2. Approche Non-Supervisée
3. Approche Supervisée
4. Conclusion

Création d'un Bag of Words

Imporation des données

- Commande SQL:
  - SELECT * FROM posts WHERE AnswerCount > 0

- Au sein de chaque texte:
  - Concatenation des paragraphes (...)
  - Suppression des liens url (<a href=...)
  - Suppression du code (<code>...</code>)
  - Suppression des symboles de mise en forme (, , <s>)

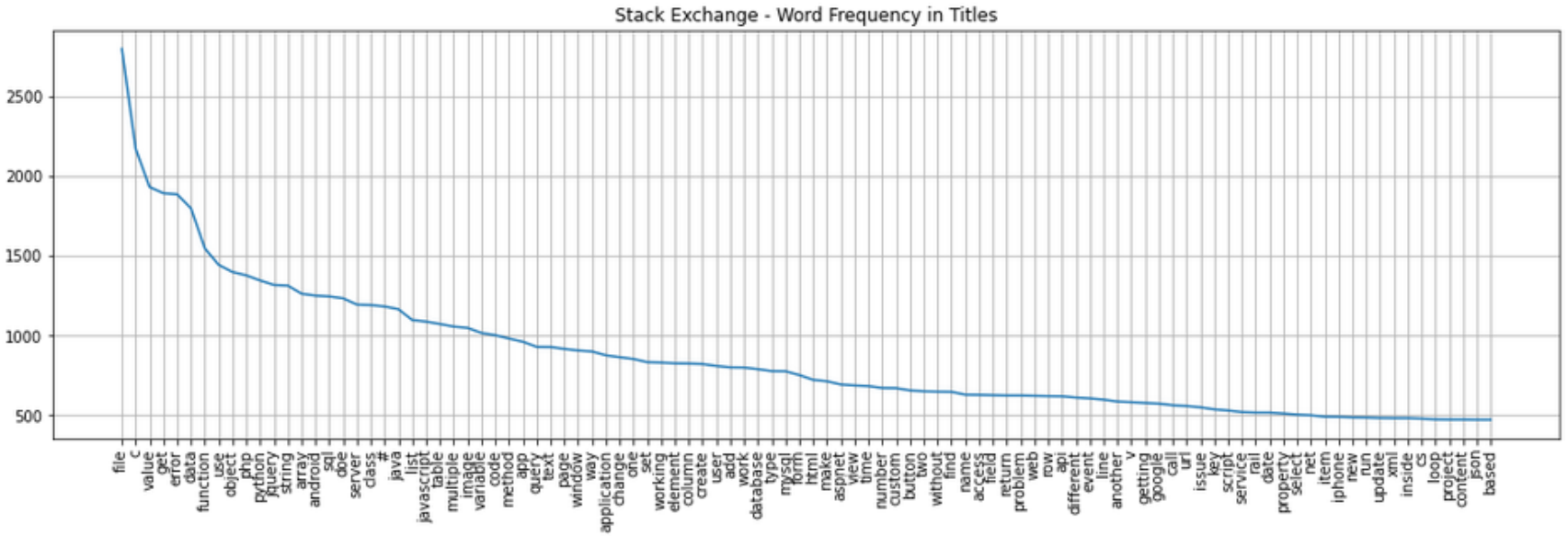
Création des bags of words (1/2)

- Définition de fonctions:
  - Pour supprimmer les caractères inutiles (-,_,+,’,",[,|,...)
  - Pour créer un Bag of Words (Tonkenize+Lemmatize)

- Création d’un Bag of Words:
  - Pour chaque Titre
  - Pour chaque Texte
  - Pour l’ensemble des Titres concaténés
  - Pour l’ensemble des Textes concaténés

Création des bags of words (2/2)

- Suppression des Stop Words
- Identification des features des questions:
  - 500 mots les plus fréquents dans les titres



- Je supprime dans les Bags of Words tous les mots qui ne sont pas les features

Approche Non-Supervisée

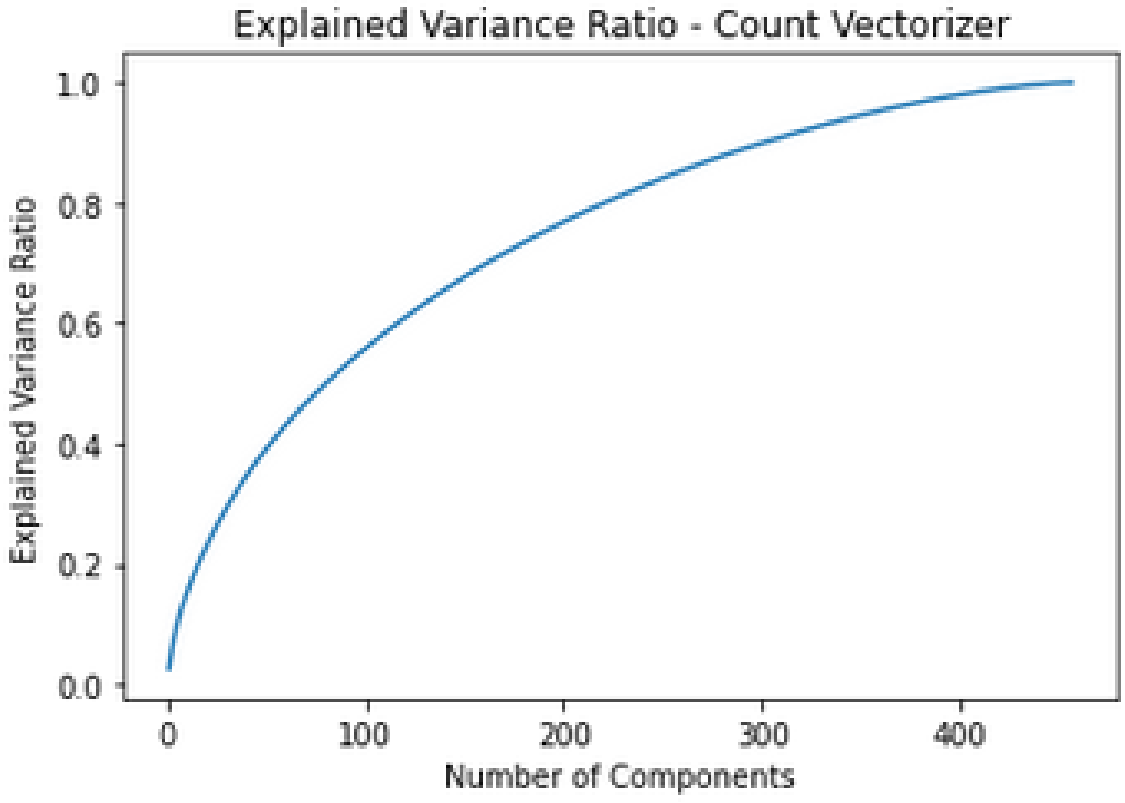
Count Vectorizer

- Création d'une liste de listes:
  - Liste avec le Bag of Words de chaque question
- Application d'un Count Vectorizer à cette liste de bag of words
- Sauvegarde dans une base de données:

	access	accessing	action	activity	add	...	write	writing	wrong	xcode	xml
10	0	0	0	0	0	...	1	0	0	0	0
11	0	0	0	0	0	...	0	0	0	0	0
12	3	2	0	0	0	...	0	0	0	0	2
13	0	0	0	0	0	...	0	0	0	0	0
14	0	0	0	0	1	...	0	0	1	0	0
15	0	0	0	0	0	...	0	0	1	0	0
16	0	0	0	0	0	...	0	0	0	0	0
17	0	0	0	0	0	...	0	0	0	0	0
18	0	0	0	0	0	...	1	0	0	0	0
19	0	0	0	0	0	...	0	0	0	0	0

Réduction des dimensions : PCA

- Application d'une PCA:



- Il faut un grand nombre de dimensions pour ne pas perdre trop de Variance:
  - PCA n'est pas optimal dans ce cas

Latent Dirichlet Allocation (LDA)

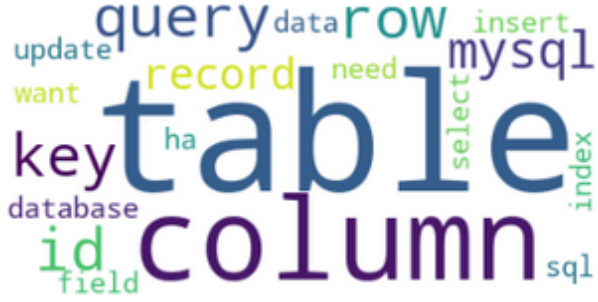
- Application d'une LDA avec 20 "topics" sous-jacents

	0	1	2	...	17	18	19
10	0.002941	0.002941	0.002941	...	0.002941	0.002941	0.002941
11	0.001429	0.001429	0.001429	...	0.001429	0.001429	0.001429
12	0.000926	0.000926	0.000926	...	0.000926	0.000926	0.000926
13	0.003125	0.003125	0.003125	...	0.352815	0.003125	0.003125
14	0.099461	0.002381	0.002381	...	0.002381	0.213452	0.410126
15	0.001667	0.601120	0.001667	...	0.001667	0.001667	0.001667
16	0.005000	0.005000	0.005000	...	0.005000	0.005000	0.005000
17	0.243455	0.161859	0.002941	...	0.293837	0.002941	0.002941
18	0.002778	0.002778	0.002778	...	0.002778	0.002778	0.002778
19	0.002778	0.002778	0.002778	...	0.002778	0.002778	0.002778

- Chaque question est localisée plus ou moins proche de chacun des 20 topics
  - Ex: Le "topic" le plus proche de question 15 est le topic 1 (0.601120 / 1)

(a) Topic 0

(b) Topic 1



(c) Topic 2

(d) Topic 15

Figure: Mots les plus courrants dans différents topics

Générer des Tags de manière non-supervisée

- Definition de 2 fonctions:
  1. Pour mapper un texte et son titre dans l'espace du Count Vectorizer
  2. Pour mapper l'espace du Count Vectorizer dans l'espace du LDA

- La seconde fonction génère ensuite des tags:
  - Identification des topics les plus proches de la question
  - Attribution des mots utilisés dans la question, et qui sont fréquents dans les topics proches.

API de Tags Non-Supervisée

Approche Supervisée

Création d'une base de données

- Identification des Tags utilisés au moins 100 fois
- Création d'une base de données:
  - 500 features
  - 1 variable binaire pour chaque tag (187 variables)
- Séparation en données Trainning (70%) / Test (30%)

Prédiction des Tags

- Régressions Logistiques et Random Forest Classifier:
  - 200 régressions pour chaque model (1 par Tag)
  - "Fitting" sur les 500 features des données Training
  - Sauvegarde des "Fit" dans un dictionaire

- Evaluation des performances des 2 modèles sur Test:
  - Logit: Accuracy supérieure à 0.9 pour tous les Tags
  - Random Forest: Accuracy supérieure à 0.9 pour tous les Tags

- Création d'une fonction prédictrice de Tags:
  - Prédiction avec Logit puis avec Random Forest

API de Tags Non-Supervisée

Conclusion

Conclusion

- Cleaning des Textes des questions
- Création de Bags of Words avec 500 mots les plus fréquents

Conclusion

- Cleaning des Textes des questions
- Création de Bags of Words avec 500 mots les plus fréquents
- Prédiction des Tags Non-supervisé:
  - Count Vectorizer
  - Latent Dirichlet Allocation (LDA)
  - Attribution des tags associés aux "topics" les plus proches

Conclusion

- Cleaning des Textes des questions
- Création de Bags of Words avec 500 mots les plus fréquents
- Prédiction des Tags Non-supervisé:
  - Count Vectorizer
  - Latent Dirichlet Allocation (LDA)
  - Attribution des tags associés aux "topics" les plus proches
- Prédiction des Tags Supervisé:
  - Count Vectorizer
  - Ajout de 200 dummies (1 pour chaque tag)
  - Séparation Training / Test
  - Fitting de Logit et Random Forest sur Training
  - Prediction des Tags sur Test:
    - Accuracy Score > 0.9 sur tous les Tags

Conclusion

- Cleaning des Textes des questions
- Création de Bags of Words avec 500 mots les plus fréquents
- Prédiction des Tags Non-supervisé:
  - Count Vectorizer
  - Latent Dirichlet Allocation (LDA)
  - Attribution des tags associés aux "topics" les plus proches
- Prédiction des Tags Supervisé:
  - Count Vectorizer
  - Ajout de 200 dummies (1 pour chaque tag)
  - Séparation Training / Test
  - Fitting de Logit et Random Forest sur Training
  - Prediction des Tags sur Test:
    - Accuracy Score > 0.9 sur tous les Tags
- Création d'API supervisée et non-supervisée