Parcours Ingénieur Machine Learning

Session Mars 2021

OPENCLASSROOMS

Projet 5
Catégorisez automatiquement des questions

03/09/2021

Etudiante : QITOUT Kenza

Mentor : Maïeul Lombard

Evaluateur : Denis Lecoeuche

CONTEXTE DU PROJET

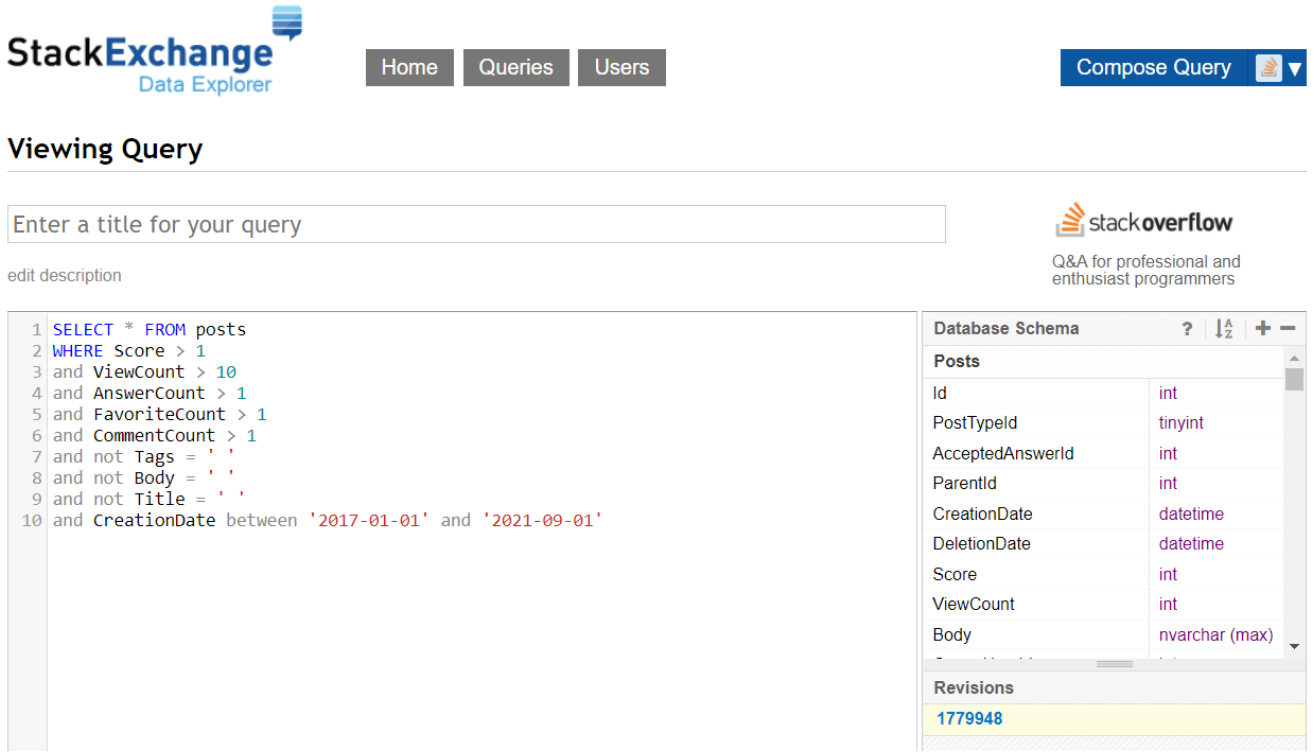
Site célèbre de questions-réponses liées au développement informatique

Poser une question —**Utiliser des tags**→ Réponse des autres utilisateurs

Objectifs : Développer un système de suggestion de tags pour le site en assignant automatiquement plusieurs tags pertinents à une question

BASES DE DONNEES

Données obtenues sur l'outil d'export [stackexchange explorer] qui recense un grand nombre de données authentiques de la plateforme d'entraide



Base de données de 50 000 lignes et 23 colonnes (fichier .csv de 12.1 Mo). Répertorie les titres, les corps et les tags associés à chaque message et les informations sur la date de création, le nombre de vues et de réponse, …

PISTES DE RECHERCHE

Missions :

- Réaliser une analyse exploratoire après avoir nettoyé le jeu de données
- Tester différents modèles
- Identifier le modèle final, le présenter dans un répertoire et le développer progressivement à l'aide d'un logiciel de gestion de versions
- Faire une API qui donnera les tags trouvés à la suite du test

Méthodologie :

- Prétraitement des messages et extraction de features avec une méthode non supervisée
- Méthode supervisée à partir des tags de Stack Overflow existants et des messages

1. NETTOYAGE ET EXPLORATION

Variables utilisées : 'Body', 'Title'

- Suppression des balises HTML avec BeautifulSoup
- Création d'une liste de stopwords (145 mots les plus fréquents + stopwords anglais)
- Suppression de la ponctuation et passage au minuscule
- Conservation uniquement des mots ['NOUN', 'PROPN']
- Tokenisation de chaque message avec différents traitements :
    - Sans traitement
    - Suppression des stopwords
    - Lemmatisation
    - Stemming
- Suppression des nombres seuls
- Suppression des espaces

1. NETTOYAGE ET EXPLORATION

Exemple : '*With xarray, how to parallelize 1D operations on a multidimensional Dataset?*'

Sans traitement :

⇨ *xarray operations dataset*

Sans les stopwords :

⇨ *xarray operations dataset*

Sans les stopwords + Lematisation :

⇨ *xarray operation dataset*

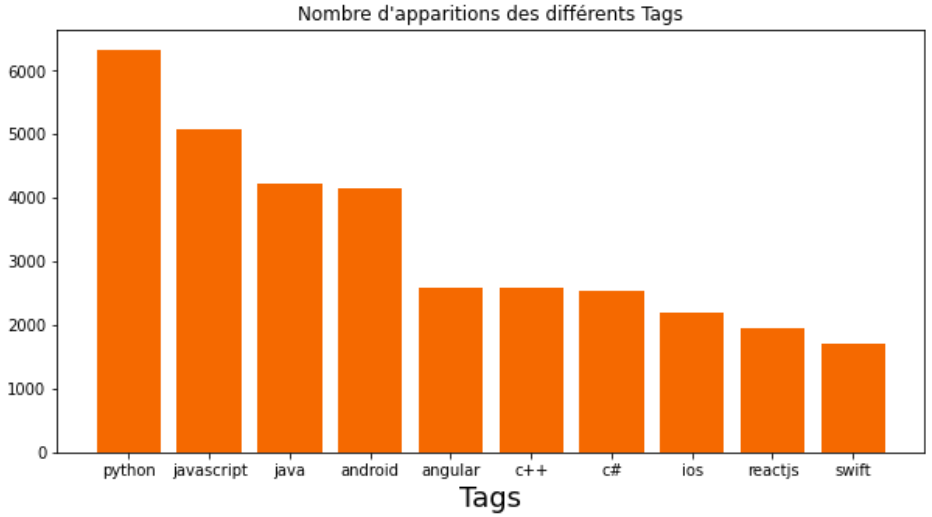
Sans les stopwords + Stemming :

⇨ *xarray oper dataset*

1. NETTOYAGE ET EXPLORATION

Variables utilisées : 'Tags'

- Suppression des symboles < et > entourant les tags
- Sélection des 100 tags les plus fréquents à conserver dans une liste (48.7% des Tags utilisés)

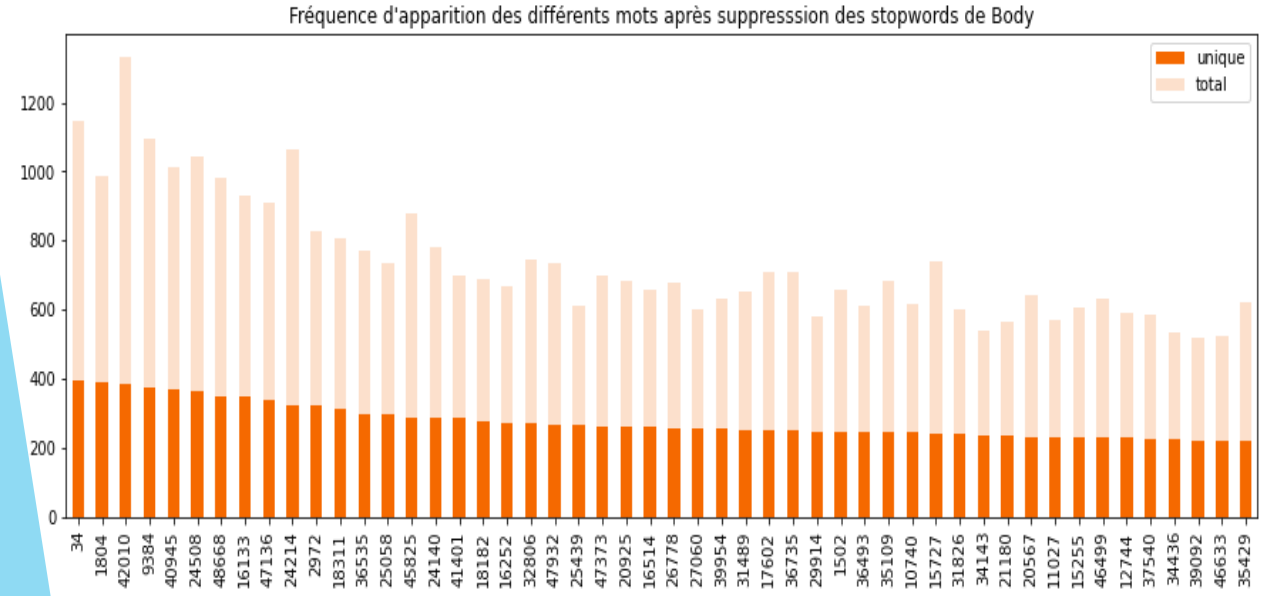


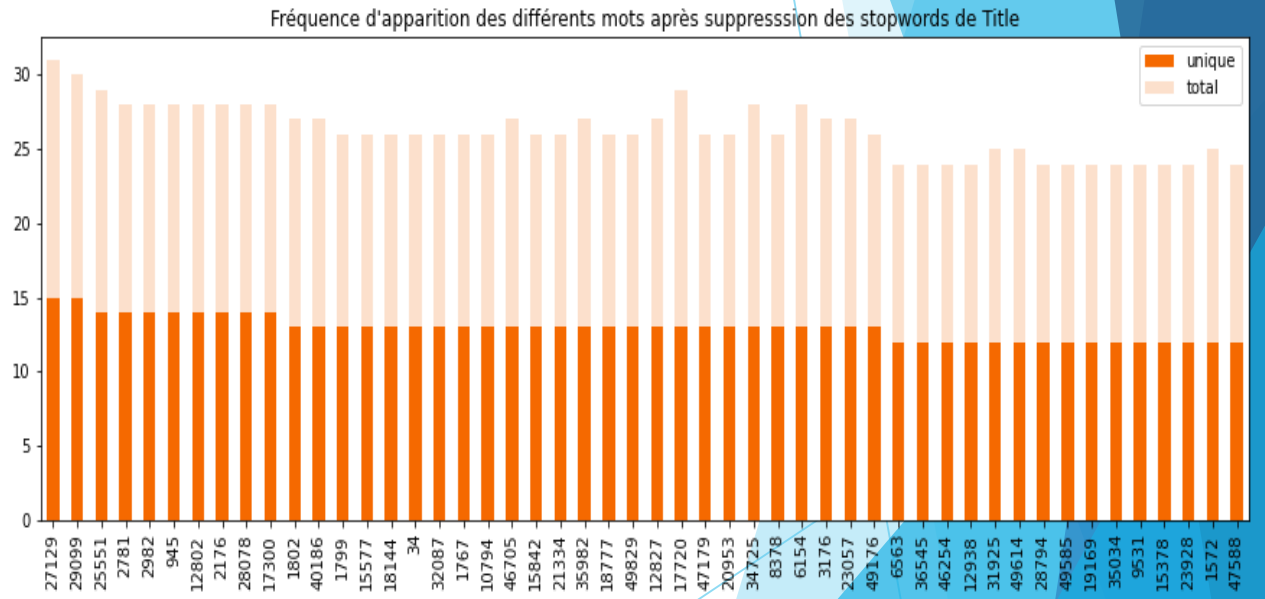
- Calcul du nombre de tags pour chaque message
- Suppression des tags ne faisant pas partie de la liste pour chaque message

<python><python-3.x><opencv3.0><opencv-contrib> → python python-3.x opencv3.0 opencv-contrib → [python python-3.x]

1. NETTOYAGE ET EXPLORATION

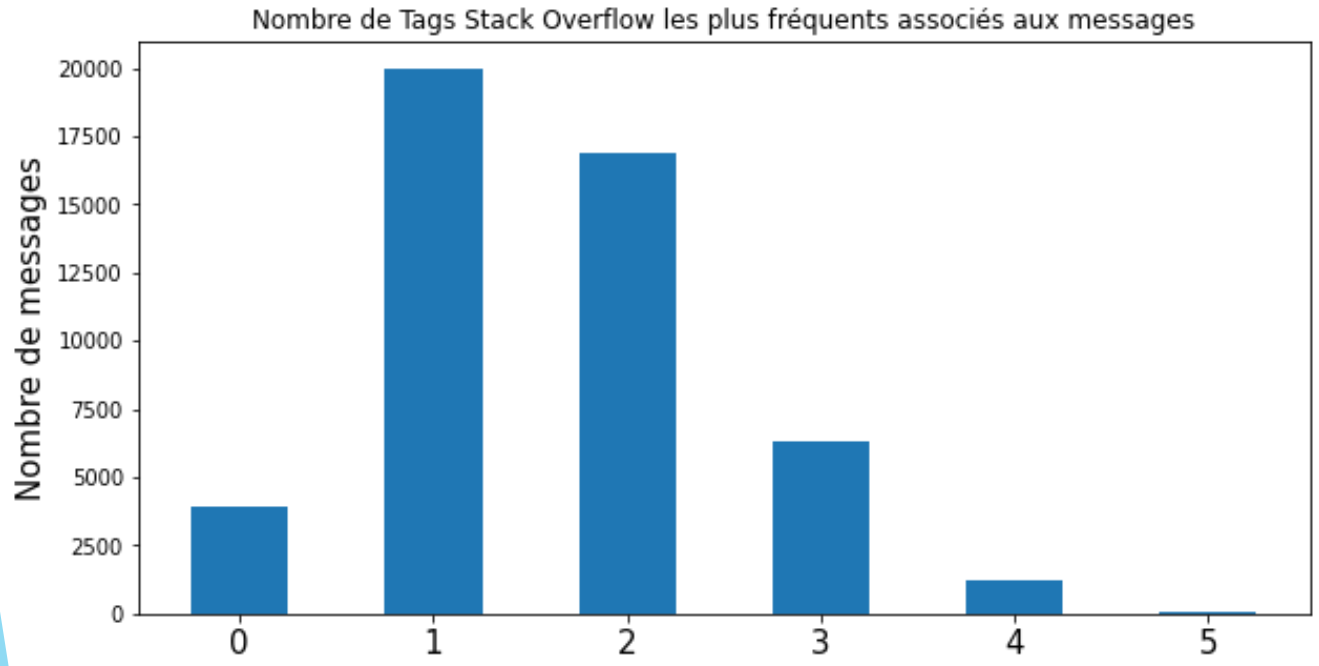
Analyse de la fréquence des mots dans les messages

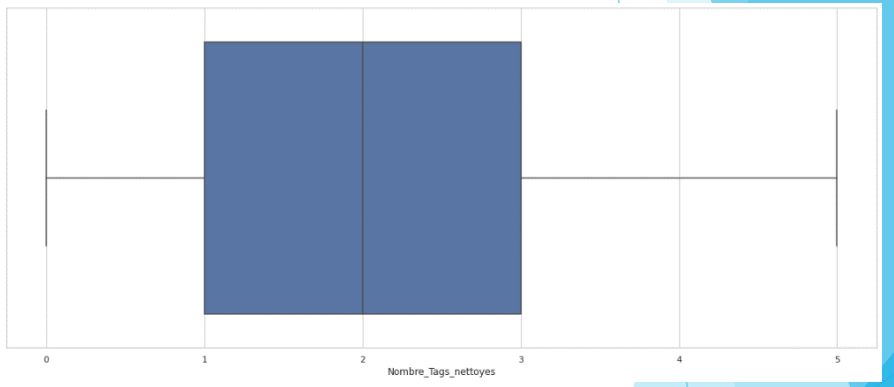




1. NETTOYAGE ET EXPLORATION

Analyse univariée du Nombre de Tags par message





Moyenne : 1.31
Médiane : 2.0
Ecart-type : 0.91
Skewness : 0.46
Kurtosis : 0.11

2. APPROCHE NON SUPERVISEE

Objectif :

Transformation des données textuelles en nouvelles features avec le modèle LatentDirichletAllocation()

⇒ Vectorization des messages avec CountVectorizer(min_df = 2, max_features = 1000)

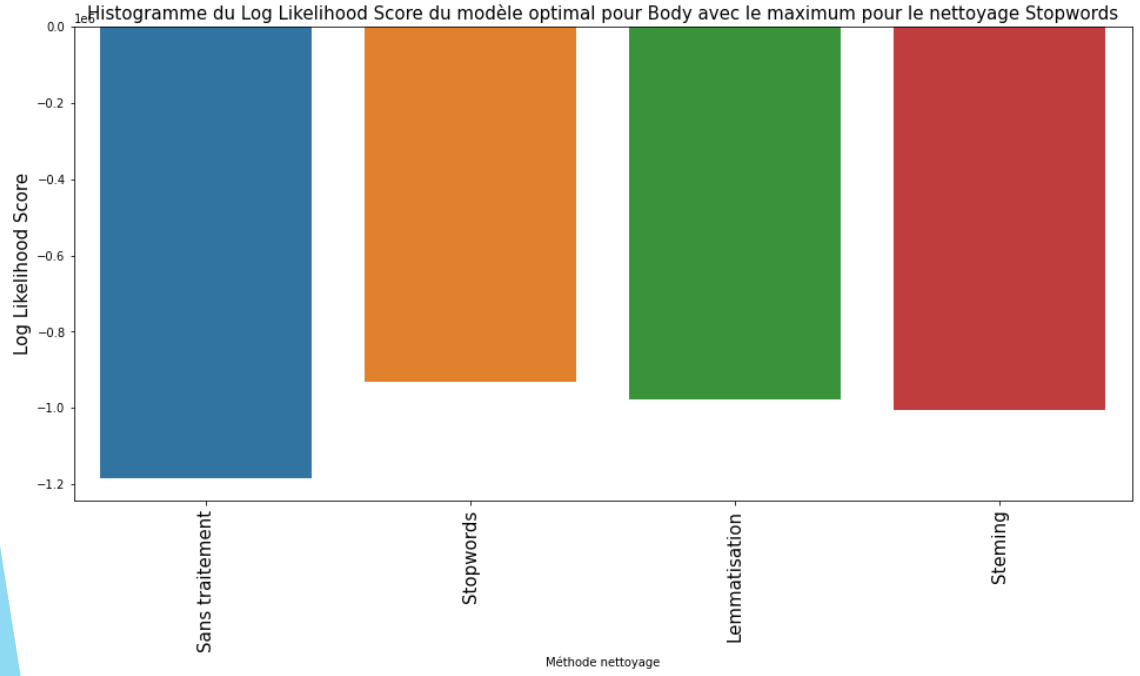
⇒ Tests sur les 4 variables créées en recherchant l'hyperparamètre optimal par validation croisée avec GridSearchCV : n_components sur [25, 50, 75, 100] et learning_decay sur [0.5, 0.8]

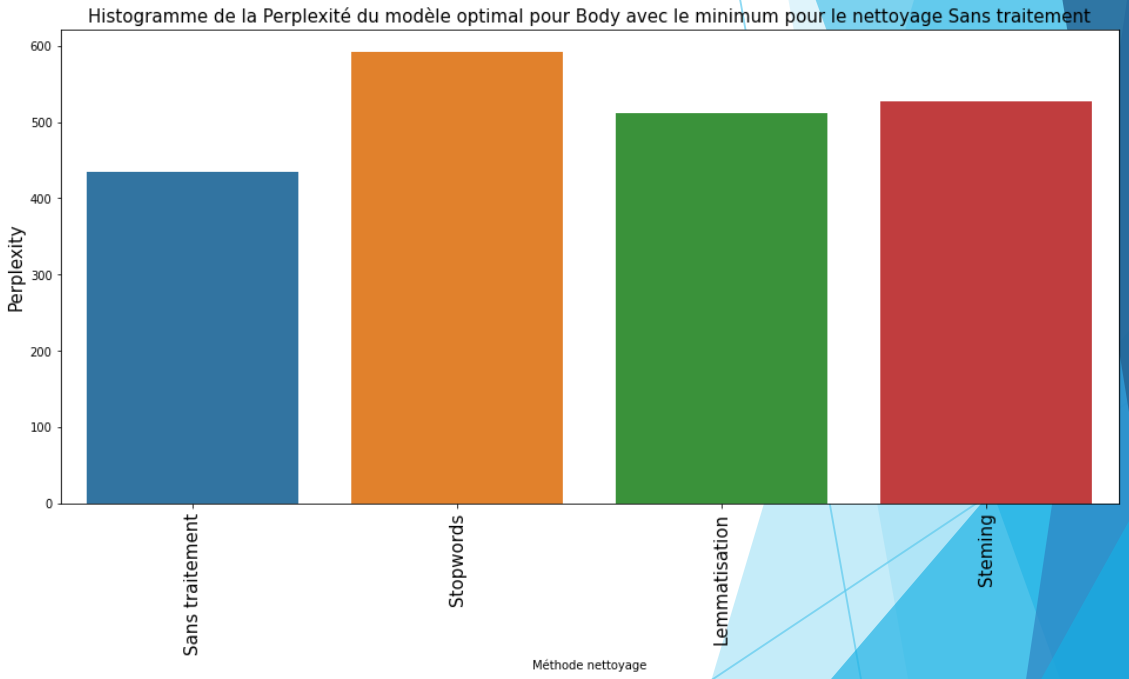
Comparaison des modèles entre eux :

- Log Likelihood Score (model.best_score_)
- Perplexity (model.perplexity(data_vectorized))

2. APPROCHE NON SUPERVISEE

Comparaison des résultats sur la variable 'Body' :



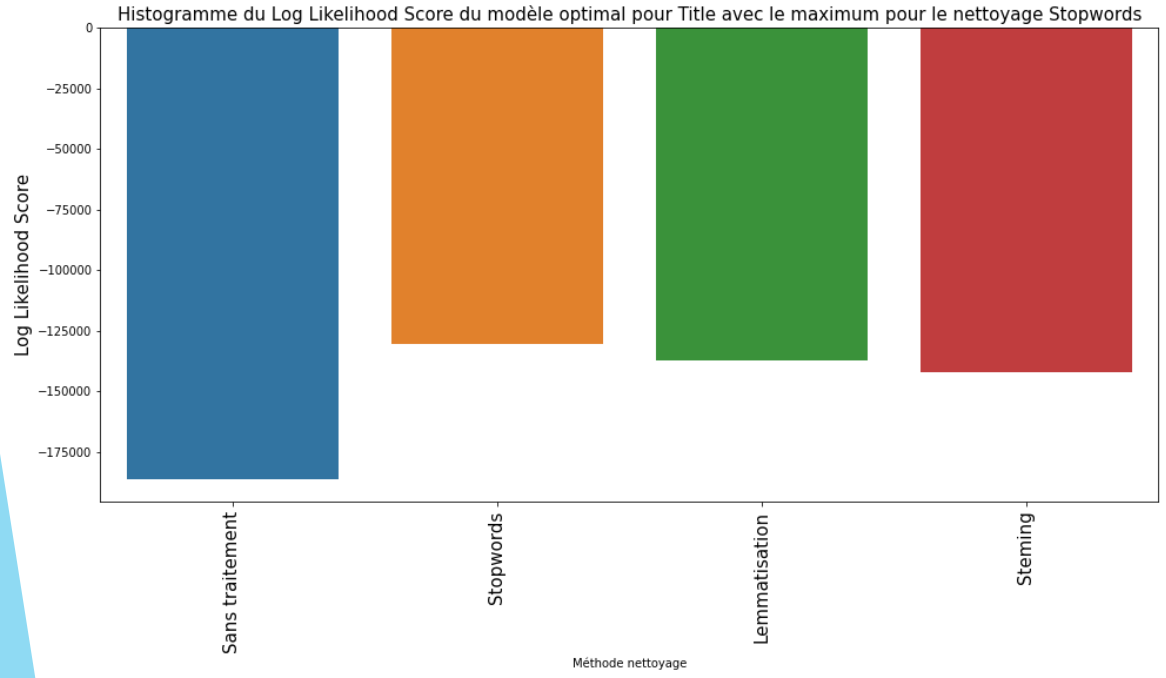


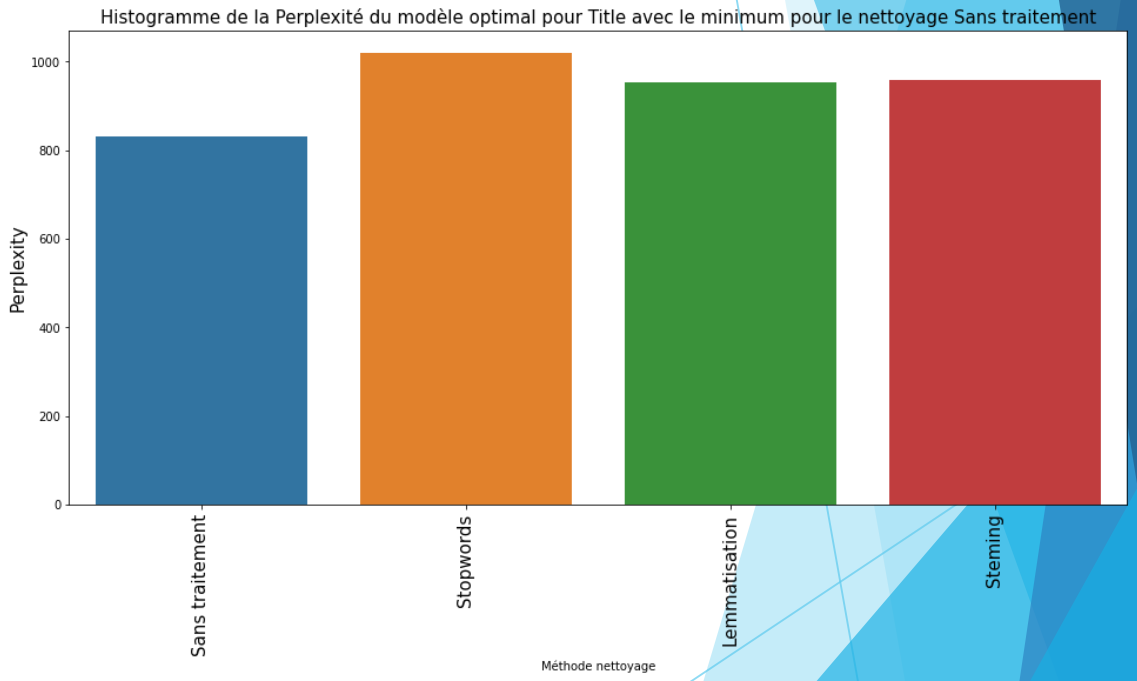
Meilleurs résultats obtenus avec le nettoyage par Stopwords (même si le nettoyage Sans traitement a une plus faible Perplexity) :

Log Likelihood Score = -9.308151e+05 et Perplexity = 591.465160

2. APPROCHE NON SUPERVISEE

Comparaison des résultats sur la variable 'Title' :





Meilleurs résultats obtenus avec le nettoyage par Stopwords (même si le nettoyage Sans traitement a une plus faible Perplexity) :

Log Likelihood Score = -130638.152465 et Perplexity = 1019.085413

2. APPROCHE NON SUPERVISEE

Sélection et création des nouvelles features :

Topics obtenus avec le modèle LDA sur la variable 'Body' avec le nettoyage Stopwords (25 Topics) =

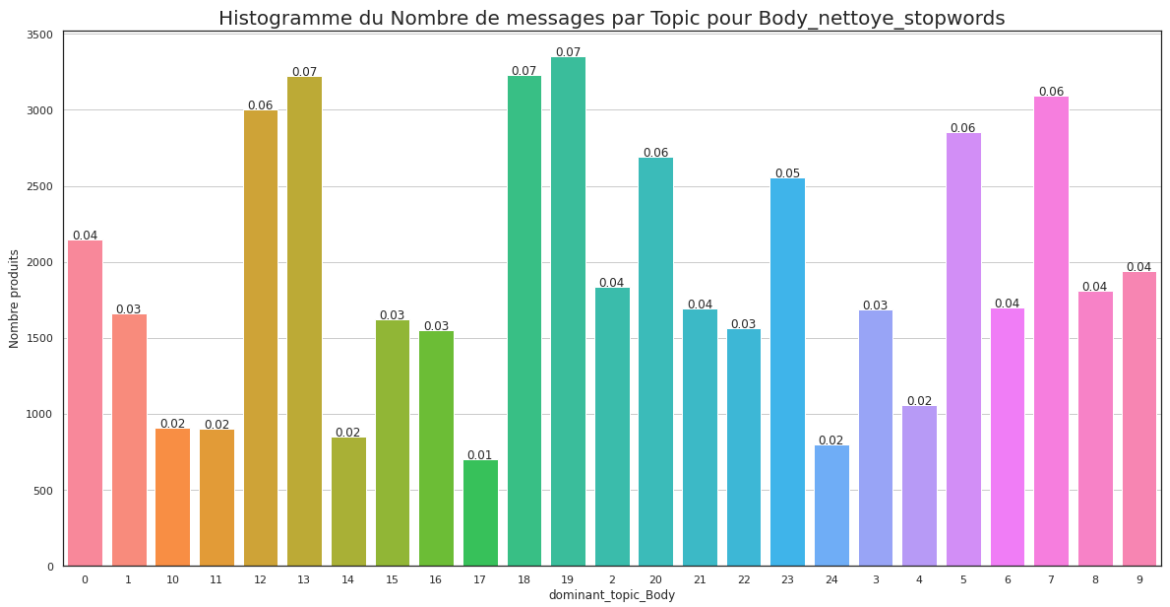
```
Topic 0:
css columns variables pointer interface methods pipeline point style frame
Topic 1:
material keras vue store email warning s3 router implementation vscode
Topic 2:
bar images client components child kubernetes video azure integer route
Topic 3:
property typescript parameter parameters position null recyclerview length angular2 login
Topic 4:
script url model argument selenium strings apps branch bit limit
Topic 5:
service input web nodejs screen system algorithm icon pip body
```

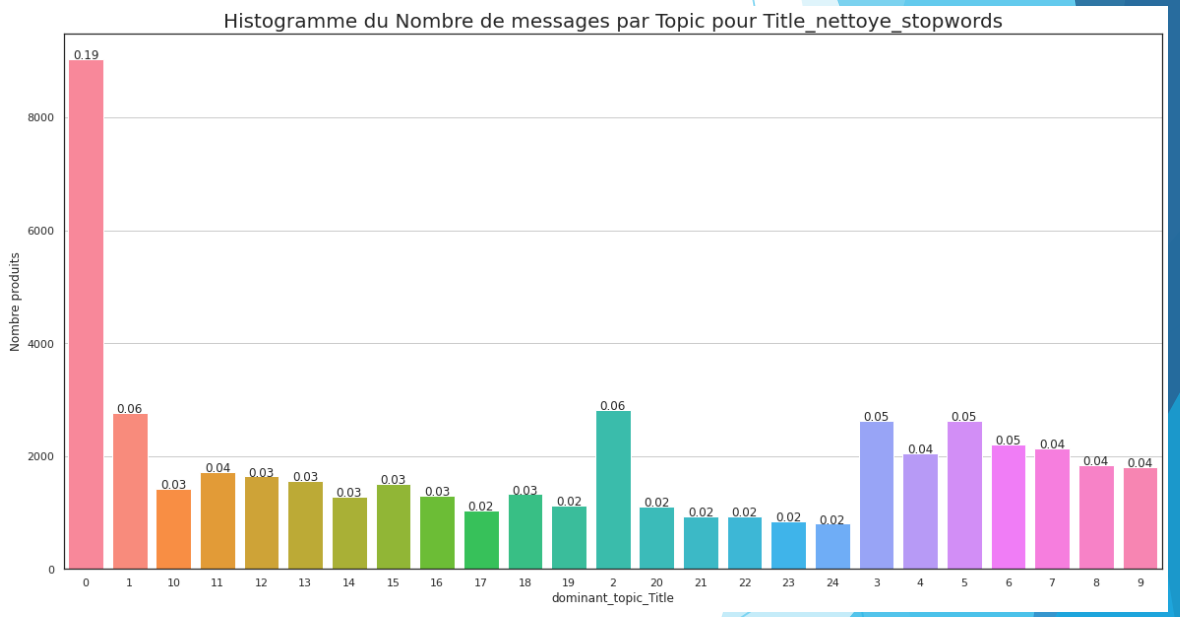
- Création des variables du poids de chaque topic pour chaque message et du Topic dominant (topic avec le poids maximal)
- Matching entre le top 20 des mots du Topic dominant et les mots contenus dans le message
- Matching entre le top 20 des mots de tous les Topics et les mots contenus dans le message

→ DataFrame final de 48 447 lignes et de 28 colonnes

2. APPROCHE NON SUPERVISEE

Analyse univariée de 'dominant_topic_Body' et 'dominant_topic_Title' :

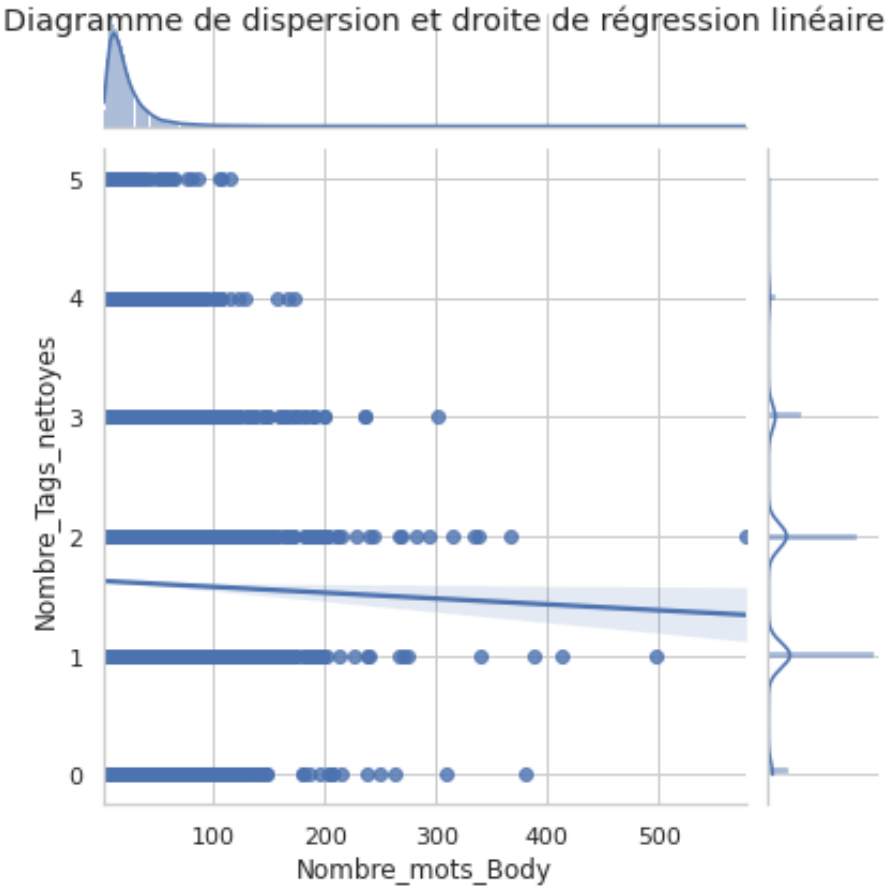




25 Topics avec le Topic 19 le plus fréquent pour la variable 'Body_nettoye_stopwords' et le Topic 0 pour la variable 'Title_nettoye_stopwords'

2. APPROCHE NON SUPERVISEE

Relation entre 'Nombre_mots_Body' et 'Nombre_Tags_nettoyes'

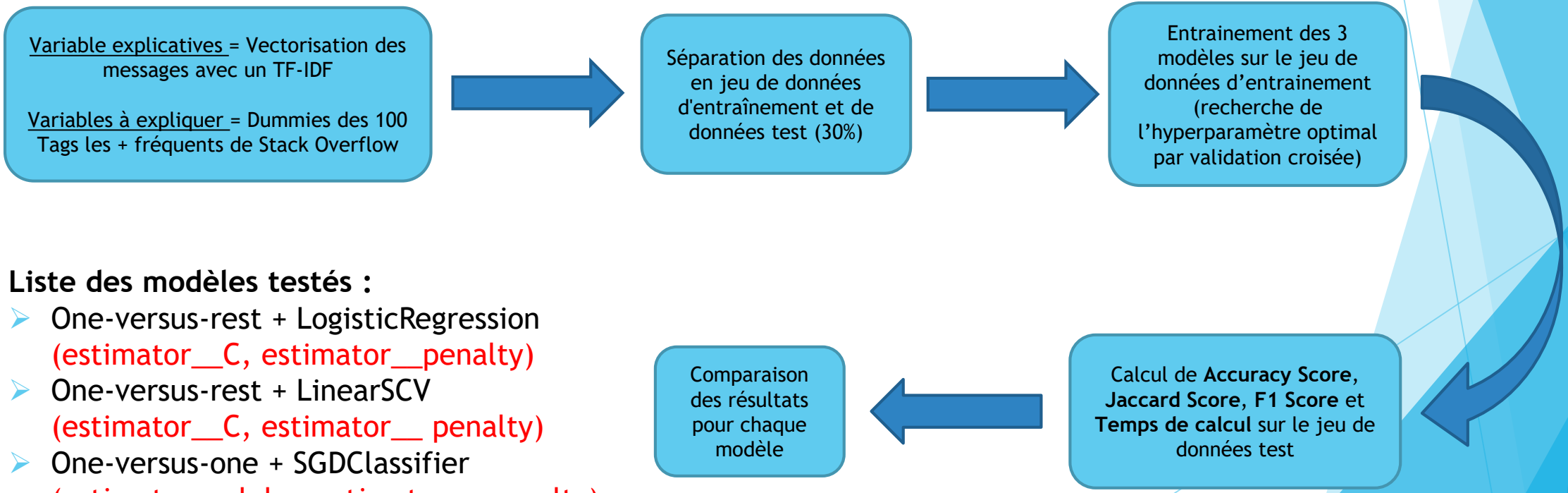


Corrélation de Pearson :
-0.059

3. APPROCHE SUPERVISEE

Objectif :

Classification multi-classe pour prédire les Tags de Stack Overflow existants à partir des messages nettoyés par le traitement Stopwords

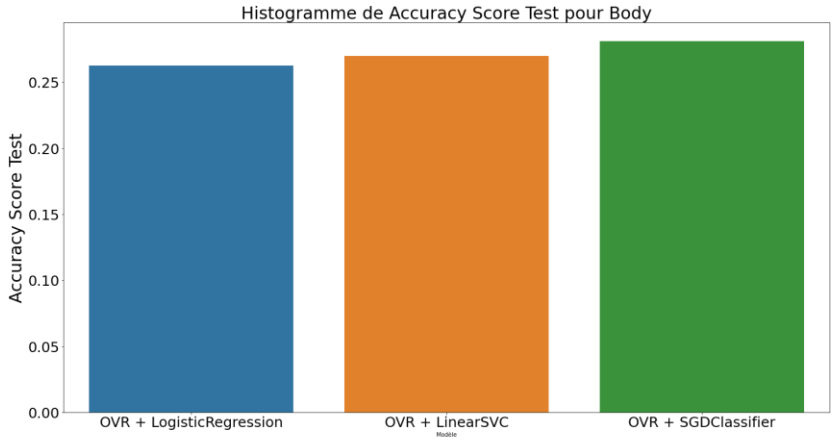


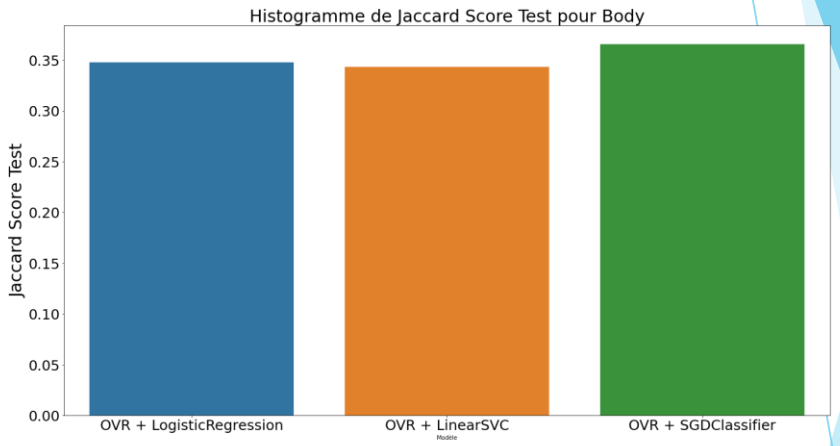
Liste des modèles testés :

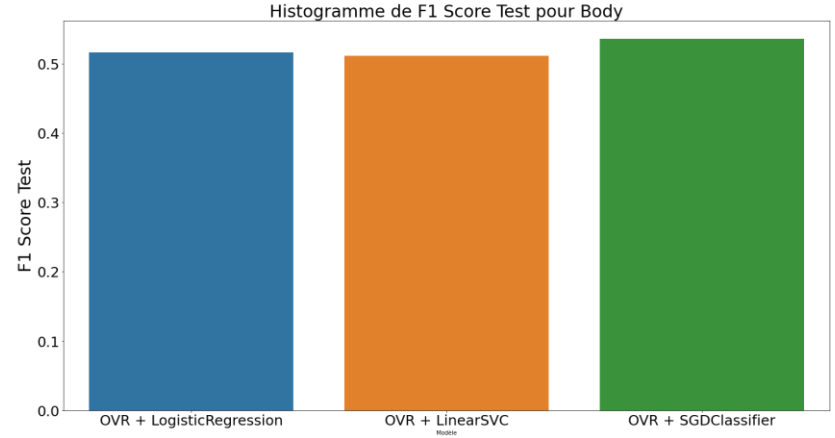
- One-versus-rest + LogisticRegression (estimator__C, estimator__penalty)
- One-versus-rest + LinearSCV (estimator__C, estimator__ penalty)
- One-versus-one + SGDClassifier (estimator__alpha, estimator__ penalty)

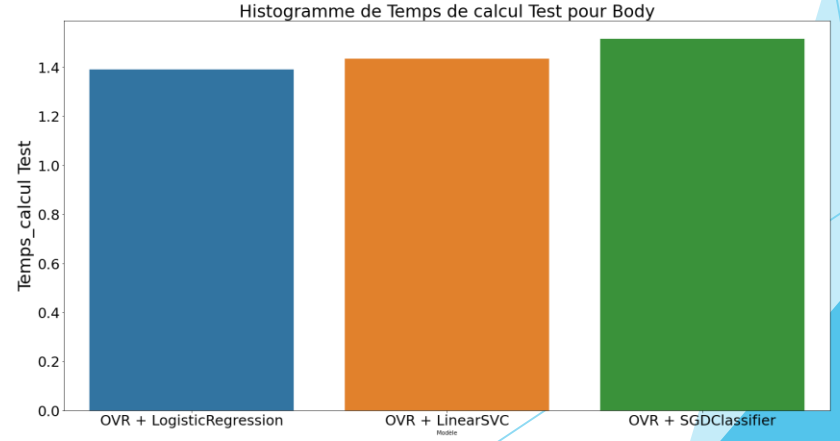
3. APPROCHE SUPERVISEE

Comparaison des résultats pour la variable 'Body_nettoye_stopwords'





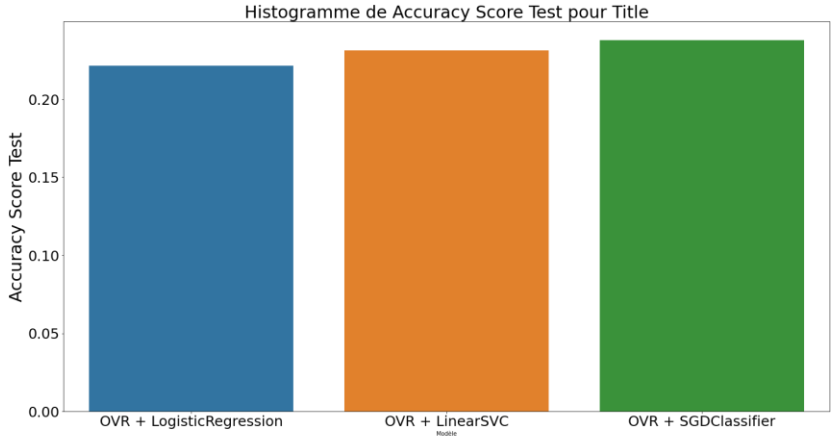


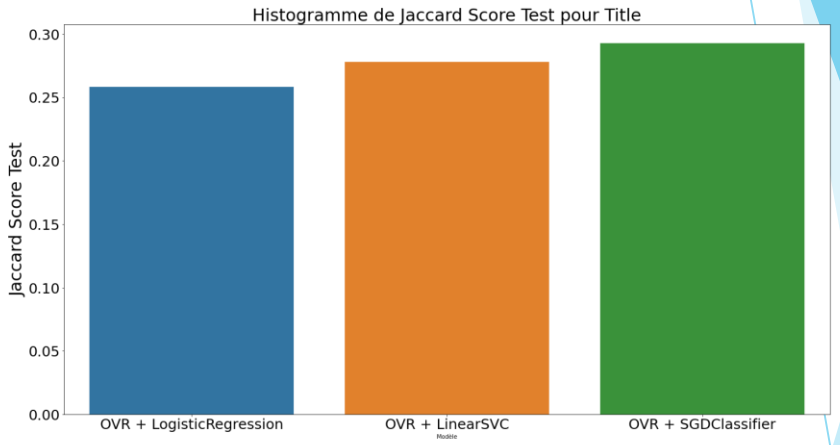


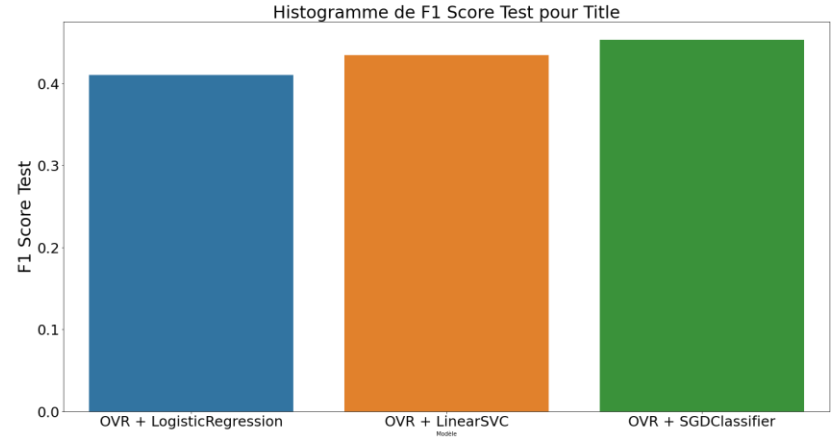
Meilleurs résultats avec le modèle SGDClassifier avec un Accuracy score (0.28), un Jaccard score (0.37) et un F1 score (0.54) légèrement plus élevés

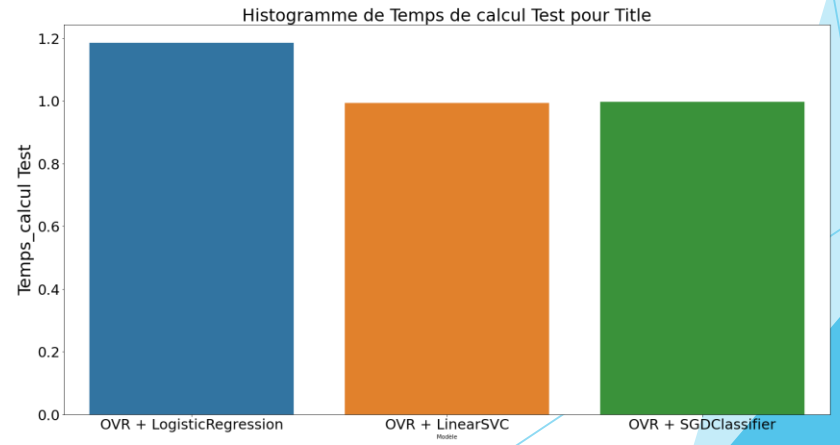
3. APPROCHE SUPERVISEE

Comparaison des résultats pour la variable 'Title_nettoye_stopwords'









Meilleurs résultats avec le modèle SGDClassifier avec un Accuracy score (0.24), un Jaccard score (0.29) et un F1 score (0.44) légèrement plus élevés

CONCLUSIONS

Objectif du projet : Développer un système de suggestion de Tags (API)

Modèles sélectionnés :

- Approche non supervisée : Tags dominants (matching avec les mots du Topic dominant) et Tags globaux (matching avec les mots de tous les Topics)
- Approche supervisée : SGDClassifier

Approche supervisée + performante (meilleures prédictions des Tags)

Améliorations possibles :

- Dictionnaire de mots contenus dans les messages avec les Tags Stack Overflow
- Word embeddings et du Deep Learning

MERCI DE VOTRE ATTENTION

QUESTIONS – REPONSES