PROJET VALIDÉ

Catégorisez automatiquement des questions

60 heures

Mis à jour le mercredi 13 mai 2020

Mise en situation

Stack Overflow est un site célèbre de questions-réponses liées au développement informatique. Pour poser une question sur ce site, il faut entrer plusieurs tags de manière à retrouver facilement la question par la suite. Pour les utilisateurs expérimentés, cela ne pose pas de problème, mais pour les nouveaux utilisateurs, il serait judicieux de suggérer quelques tags relatifs à la question posée.

Amateur de Stack Overflow, qui vous a souvent sauvé la mise, vous décidez d'aider la communauté en retour. Pour cela, 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.

Les données

Stack Overflow propose un outil d'export de données - "stackexchange explorer", qui recense un grand nombre de données authentiques de la plateforme d'entraide.

Vous souhaitez réviser votre SQL pour récupérer les données ? Il y a **un cours** pour ça.

Par défaut, il y a une limite sur le temps d'exécution de chaque requête SQL, ce qui peut rendre difficile la récupération de toutes les données d'un coup. Pour récupérer plus de résultats, pensez à faire des requêtes avec des contraintes sur les `id` . Par exemple :

sql

```sql
SELECT * FROM posts WHERE Id < 50000
```

Contraintes

- Mettre en œuvre une approche non supervisée.
- Utiliser une approche supervisée ou non pour extraire des tags à partir des résultats précédents.
- Comparer ses résultats à une approche purement supervisée, après avoir appliqué des méthodes d'extraction de features spécifiques des données textuelles.
- Mettre en place une méthode d'évaluation propre, avec une séparation du jeu de données pour l'évaluation.
- Pour suivre les modifications du code final à déployer, utiliser un logiciel de gestion de versions, par exemple Git.

Livrables attendus

- **Un notebook d'exploration** comprenant une analyse univariée, une analyse multivariée, une réduction dimensionnelle et les différentes questions de recherches associées (non cleané, pour comprendre votre démarche).
- **Un notebook de test** de différents modèles (non cleané, pour comprendre votre démarche).
- **Le code final à déployer** présenté dans un répertoire et développé progressivement à l'aide d'un logiciel de gestion de version (plusieurs commits cohérents).
- **Le point d'entrée d'une API** disponible pour le test.
- Un **rapport** (en PDF) présentant les différents traitements effectués, et détaillant les modélisations effectuées, en particulier celle qui est en production.
  Ce rapport...

1. montrera vos capacités de synthèse (environ 10 pages) ;
2. respectera des règles de présentation (plan, sommaire, orthographe, mise en page, titres des figures, lisibilité des figures) ;
3. possédera un contenu scientifique : ne pas sur-simplifier (garder les noms des algorithmes, expliquer clairement la démarche).

- **Une présentation** servant de support à la soutenance.

Pour faciliter votre passage au jury, déposez sur la plateforme, dans un dossier nommé "*P5_nom_prenom*", tous les livrables du projet. Chaque livrable doit être nommé avec le numéro du projet et selon l'ordre dans lequel il apparaît, par exemple "*P5_01_notebookexploration*", "*P5_02_notebooktest*", et ainsi de suite.

Modalités de la soutenance

Votre soutenance, **auprès d'un mentor validateur,** durera 25 minutes, découpées ainsi (à titre indicatif) :

- 5 min - Présentation de la problématique, de son interprétation et des déductions effectuées quant aux pistes de recherche possibles
- 5 min - Présentation du cleaning effectué, du feature engineering et de l'exploration
- 10 min - Présentation du modèle final sélectionné ainsi que des améliorations effectuées
- 5 min - Présentation du modèle final sélectionné ainsi que des performances et améliorations effectuées
- 5 à 10 minutes de questions-réponses

Ressources complémentaires

- Excellent tutoriel pour maîtriser quelques concepts de base du traitement de texte, notamment les bags of words. De manière générale, pensez à consulter la documentation de scikit-learn !
- Article récapitulatif sur une famille de méthodes pour assigner des mots-clés à un texte. Lisez cet article pour avoir une vue d'ensemble des méthodes existantes. Attention, cet article, en plus de la vue d'ensemble, présente une famille de méthodes plus compliquées qui vont au-delà des exigences de ce projet.
- Tutoriel du site Kaggle sur le traitement du texte en Python. Il est très complet, et traite aussi des aspects annexes (preprocessing, etc.) avec des exemples de code.
- Version en ligne du livre "Natural Language Processing with Python", qui traite de l'utilisation de la librairie NLTK en Python.
- Une présentation des méthodes non supervisées de modélisation de thèmes.

Compétences évaluées

- Représenter graphiquement des données à grandes dimensions
- Prétraiter des données non structurées pour obtenir un jeu de données exploitable
- Mettre en œuvre des techniques d'extraction de features pour des données non structurées
- Mettre en œuvre des techniques de réduction de dimension

FAQ

POUR LES ENTREPRISES

Former et recruter

EN PLUS

Boutique

Mentions légales

Conditions générales d'utilisation

Politique de protection des données personnelles

Cookies

Accessibilité

Français

Télécharger dans l'App Store