Rapport de projet

Fouilles de données

PARIS DIDEROT

Enseignant : Anne-Claire Haury

Université Paris Diderot

Master Informatique 2ème année

Quentin BOUILLAGET - Stéphane SCHMIDELY

Préface

Dans le cadre de l'UE « Fouilles de données », nous devions concevoir une application mettant en pratique les différentes notions et algorithmes utilisés en datamining.

Pour notre projet, nous sommes parti sur un logiciel capable de recommander des produits en fonction des tweets d'un utilisateur. Ce document a pour but d'expliquer le problème soulevé, nos motivations ainsi que les différents moyens mis en place pour le résoudre.

Introduction

L'application que nous avons mis en place souhaite répondre à la problématique suivante : « Pouvoir proposer à un utilisateur des produits qui lui correspondent ». Nous voulions pouvoir permettre à un utilisateur de lui recommander différents produits basés sur le contenu qu'il publie, qu'il aime ...

Méthodes

Base de données

Etant donnée que l'application est en temps-réel, nous n'avons pas eu besoin de créer une base de données. Par conséquent, aucune information n'a été collectée et sauvegardée. Tous les traitements que nous sommes amenés à faire dans notre application résulte du choix de l'utilisateur à un instant donné. En effet, le contenu dépendra de l'utilisateur selectionné mais aussi du contenu publié par ce dernier au moment de la requête.

Langage et librairies

Nous avons choisi de développer l'application en Python notamment dû au fait qu'il dispose d'un certains nombre de librairies dédié au datamining. Par ailleurs, ce projet était aussi une occasion pour nous d'apprendre ou de pratiquer de nouveau ce langage. Enfin, il nous a permis de découvrir un nouveau framework Web : Django.

Lors du développement de notre application, nous avons eu recours à plusieurs librairies externes :

Amazon Product : https://pypi.python.org/pypi/python-amazon-product-api/

Hunspell : https://pypi.python.org/pypi/hunspell

Tweepy : https://github.com/tweepy/tweepy

Amazon Product

Hunspell

Nous avons décidé d'avoir recours à cette librairie car elle permet entre autres de vérifier si des mots existent dans les dictionnaires français et anglais, de proposer des synonymes et de donner la racine d'un mot dans le but d'uniformiser et de corriger les données que nous allions recevoir en entrée.

Par ailleurs, Hunspell est une librairie utilisée dans de nombreux programmes comme LibreOffice, Firefox, Thunderbird, Google Chrome ce qui nous a conforté dans notre choix.

Tweepy

Tweepy est une des nombreuses librairies disponible en Python pour interfacer avec l'API de Twitter. Nous avons décidé de la choisir en raison de sa popularité mais aussi grâce à sa facilité d'utilisation. Elle nous permet de récuperer les différents tweets d'un utilisateur donné mais aussi lors de la saisie du nom du compte Twitter afin de lui suggérer différents profils associés à sa saisie.

Traitements

La seule action requise par l'utilisateur est de saisir le nom du compte Twitter dont il souhaite connaître les différents produits associés à ses contenus. Une fois la validation effectuée, le programme se charge de :

- récupérer les 100 derniers tweets du profil choisi
- nettoyer chacun des tweets
  - supprimer les URL
  - supprimer la ponctuation
  - supprimer le # des hashtags
  - supprimer les références au profil (@compte)
  - supprimer les prépositions/stop-words (le, la, du ...)
  - convertir le tout en minuscule
  - corriger les mots et remplacer par la racine de chaque mot
- séparer le corps du tweet, les hashtags
- création d'une liste de 'tokens'
- calcul du TD-IDF sur l'ensemble des tokens afin de déterminer les mots-clés majoritaires avec un point plus fort sur les mots venant des hashtags
- requête vers l'API d'Amazon pour récuperer une liste de produits en lien avec les mots-clés calculés précédemment

La majeure partie de l'application se situe au niveau du traitement des tweets afin de pouvoir ensuite calculer les mots-clés au plus proche du contenu publié par le compte twitter sélectionné.

Résultats

Discussion

Conclusion

Ce projet nous a permis de mettre en pratique les différentes notions et méthodes de datamining vu en cours appliqué à un cas concret. Nous avons du faire face à différents problèmes soulevés par les grandes quantités de données réçus notamment au niveau du contenu des tweets de chaque utilisateur.

Il nous a aussi permis de découvrir plus en profondeur Python et d'utiliser différentes librairies en lien avec les API de Twitter et d'Amazon.

Ce logiciel possède un grand nombres d'extensions possibles en commençant par la possibilité de récuperer du contenu d'un utilisateur depuis son compte Facebook. Il existe encore bien d'autres ajouts imaginables qui permettraient de rendre l'application encore plus précise et plus proche des recommandations que le site web pourrait leur faire.

Equipe

Répartition des tâches

Dans le cadre de notre projet, nous devions avoir recours à deux API : celle de Twitter afin de récuperer les différents tweets d'un utilisateur et Amazon pour qu'il nous renvoie une liste de produits associées aux différents mots clés que nous lui fournissions en paramètre.

Stéphane s'est chargé de mettre en place l'interface avec l'API Twitter mais aussi de nettoyer et « harmoniser »les différents tweets de l'utilisateur afin de pouvoir ensuite récuperer des mots-clés à soumettre à Amazon. De son côté, Quentin s'est occupé de l'API Amazon et du TD-IDF sur les mots clés reçus.

Le site web a été développé par nous deux.

Listes des figures