François d'Anselme (faolin sur Git)
Grégoire de Charnace(gregoireDC sur Git)
Philippe Kreiss

Projet R :

Introduction :

Dans ce projet nous allons construire un système de recommandation avec le logiciel R le tout basé sur une technique d'analyse de sentiment.

a) Choix du Dataset :

Nous sommes partis d'un dataset d'Amazon. IL s'agit du dataset « clothing, shoes and jewellery. Nous avons pris la version sans les notes.

En effet nous allons chercher à créer un système de note base sur de l'analyse de sentiment et pour cela il nous fallait un dataset avec un ID utilisateur, un ID produit et la review texte de l'utilisateur sur laquelle nous allons effectuer notre analyse de sentiment.

b) Choix du lexique de sentiment :

Pour l'analyse de sentiment nous avons voulus tester 2 approches relativement différentes, une approche moins précise et plus efficace de type positif/négatif, une autre plus précise mais moins efficace en terme de notation.

Pour le premier lexique de sentiment nous utiliserons « Bing » car comme il est dit dans la conclusion du texte « a pair of text analysis explorations » c'est le lexique qui donne les résultats les plus efficaces pour estimer la positivité ou la négativité d'un texte. Ce lexique est moins précis car il ne permet pas d'exprimer des variations dans les émotions. Il est cependant efficace car il ne comporte que 2 classes de sortie et moins il y a de classe plus la notation est facile donc efficace.

Pour le 2éme lexique de sentiment nous utiliserons « nrc » car il nous permettra d'avoir un résultat plus précis que seulement du binaire avec également les émotions qu'éprouve l'utilisateur lors de l'écriture de sa review. Cependant dût à ses multiples classes, nrc risque d'être moins efficace, de plus il est beaucoup moins intéressant de l'utiliser dans un système de recommandation.

Il sera néanmoins intéressant de comparer les résultats de ces 2 lexiques de sentiments puisque « nrc » est censé être noté plus positivement que Bing mais également converge plus facilement et rapidement vers le sens global de la review.

c) Prétraitement des données :

Comme préciser précédemment nous avons utilisé un dataset amazon, ce dataset est une archive au format .zip . Afin de récupérer le contenu de l'archive et de le passer directement en .csv nous avons utilisé un script python (disponible sur le git) .Nous créons donc un nouveau fichier nommé « data_transformer.csv » que nous allons exploiter.

Pour commencer nous lisons le csv que nous chargeons dans une dataframe « df » . Le volume de données étant important (280 000 reviews) nous avons diminué la taille du dataset (

afin d'opitmiser la vitesse de notre script R et d' éviter de saturer la mémoire de notre pc) . Ce nouveau dataset est chargé dans une nouvelle dataframe « df1 » .

Une fois que nous avons récupérer un jeu de données exploitable nous avons commencé le prétraitement pour le sentiment anaylisis .

Nous nous sommes appuyés sur la bibliothèque « tidytext ». Afin de pouvoir anaylser chaque mot séparément nous avons transformé le dataset de manière à ce qu'il n'ait plus qu'un seul mot par ligne.

En conclusion sur ce prétraitement on peut souligner que le dataset d'amazon était déjà presque prêt à l'emploi. Nous avons eu à choisir les champs qui nous intéressaient et à remettre en forme afin de pouvoir faire le scoring sur le sentiment analysis.

d) Sentiment analysis

Comme préciser dans l'introduction nous avons testé 2 méthodes : bing et nrc. Pour chaque méthode nous avons utilisé la librairie « tidytext ».

La première étape a été de scorer tous les mots. Pour cela nous avons utilisé les 2 méthodes sur la dataframe ne contenant qu'un seul mot par ligne.

Cependant nous n'avions pas une note globale des commentaires par produit. En effet nous avons jugés intéressant de chercher à regrouper toutes les notes des commentaires relatifs à un produit (un type de produit étant manifesté par la variable « asin » dans notre jeu de données).

Pour cela nous avons utilisé la méthode « inner join » et un « count » qui nous a permis d'avoir la note globale d'un produit en fonction de ses commentaires associés.

Pour bing nous avons obtenus une dataframe avec l'asin du produit ainsi qu'une note. Il y a un 3éme champs très important qui définit la pondération (+/-) de la note. Ainsi on peut savoir que la note 29 associée au mot négatif dans le 3éme champs corresponds a -29.

Pour nrc nous avons obtenus une dataframe presque identique à la précédente. Cependant nous avons une ligne par produit et type de sentiment exprimé. Ainsi un produit pourra avoir une note dans plusieurs sentiments différents. Grâce à ce procédé nous avons donc une vision beaucoup plus précise des sentiments exprimés par les utilisateurs.

Nous avons décidés de faire des représentations graphiques de ces différents modèles pour cela nous avons utilisés la librairie « ggplot2 ». Nous avons choisis de les représenter sous forme d'histogrammes. L'axe des abscisses représentant les différents produits ici la variable « asin ».

l'Axe des ordonnés représente pour bing la note générale du produit. En bleu/vert nous avons les produits notés positivement (ils partent donc vers le haut du graphique). En rouge nous avons les produits notés négativement (ils partent donc vers le bas du graphique).

L'axe des ordonnés représente pour nrc une échelle de notes non centrées en 0. Pour chaque produit nous avons une seule barre avec plusieurs délimitations de couleurs. Chaque couleur de la barre représente à un sentiment exprimé par l'utilisateur. Il faut donc regarder la différence

entre le bas et le haut de la couleur pour savoir la note d'un sentiment précis pour un produit. Dans ce graphique on suit la même règle que pour bing, les sentiments négatifs sont en dessous de 0 et les sentiments positifs au-dessus.

Les 2 graphiques sont disponibles en annexe et nous permettent d'avoir une excellente visibilité sur notre système de notation.

e) Système de recommandation :

Pour le système de recommandation nous avons décidés de fusionner 2 dataframe. Nous avons fusionné la dataframe contenant notre scoring par produit et celle contenant les users, afin d'avoir un système qui correspond à l'UCBF (filtrage collaboratif explicite basé sur les utilisateurs). Nous avons ensuite recréé une dataframe avec uniquement les champs qui nous intéressaient. Le but de cette manœuvre est de pouvoir intégrer notre scoring précedent dans le système de recommandation.

Pour toute notre analyse nous allons nous appuyer sur la librairie « recommenderlab » qui fournit les outils dont nous avons besoin pour créer notre système de recommandation. De plus nous n'allons effectuer notre analyse que sur notre scoring « bing » car c'est le plus approprié des 2 pour ce que nous allons faire.

Une fois ce traitement fait et donc le dataset prêt nous avons créé la realratingmatrix que nous appelons « data_mtx ». Depuis cette matrice nous créons notre système de recommandation de type « UCBF ». Nous l'appelons « r1 ».

Nous avons testé un système de prédiction grâce à la fonction « predict ». Le type de cette recommandation est ratings et nous l'avons appelée « p1 ». Nous avons donc à ce stade un système de recommandation.

f) Evaluation du système de recommandation :

Dans cette partie nous allons mettre en place des outils permettant de mesurer la précision de notre système. Pour cela nous nous appuyons sur la méthode « evaluate » de recommenderlab.

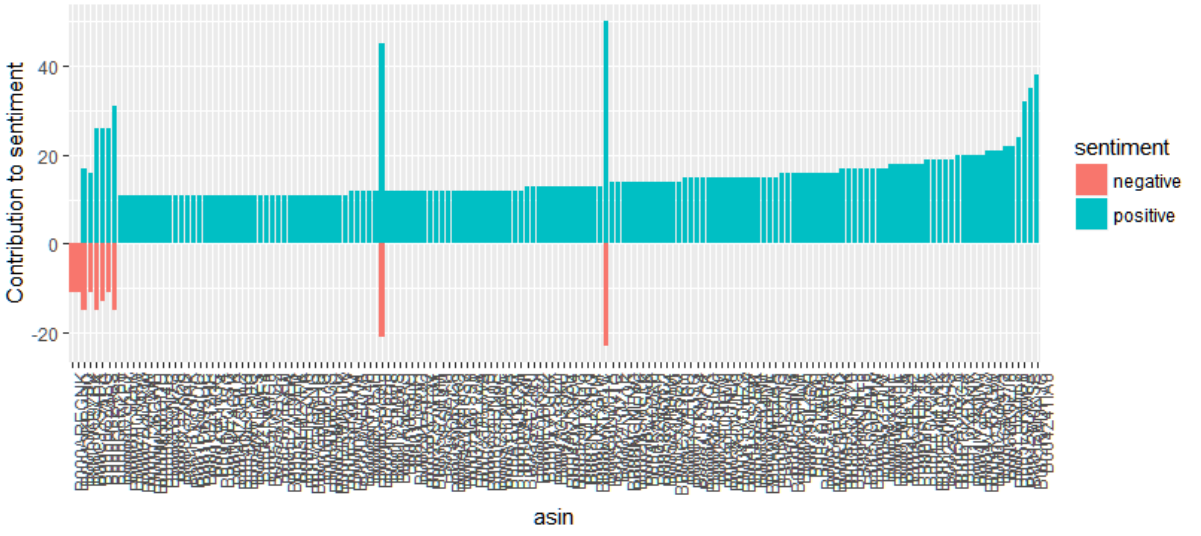
Nous avons faits des essais pour cette méthode, elle nous permets d'obtenir les « trues postives », « false postives » , « false negative », « true negative », la précision et le recall.

Tout ceci sous forme d'une grande matrice qui permet de bien visualiser l'efficacité du modèle. Cependant dans la théorie cette matrice est parfaite pour notre cas mais nous n'avons pas réussi à l'utiliser. Nous avons tout de même laissé le code que nous avons testé.

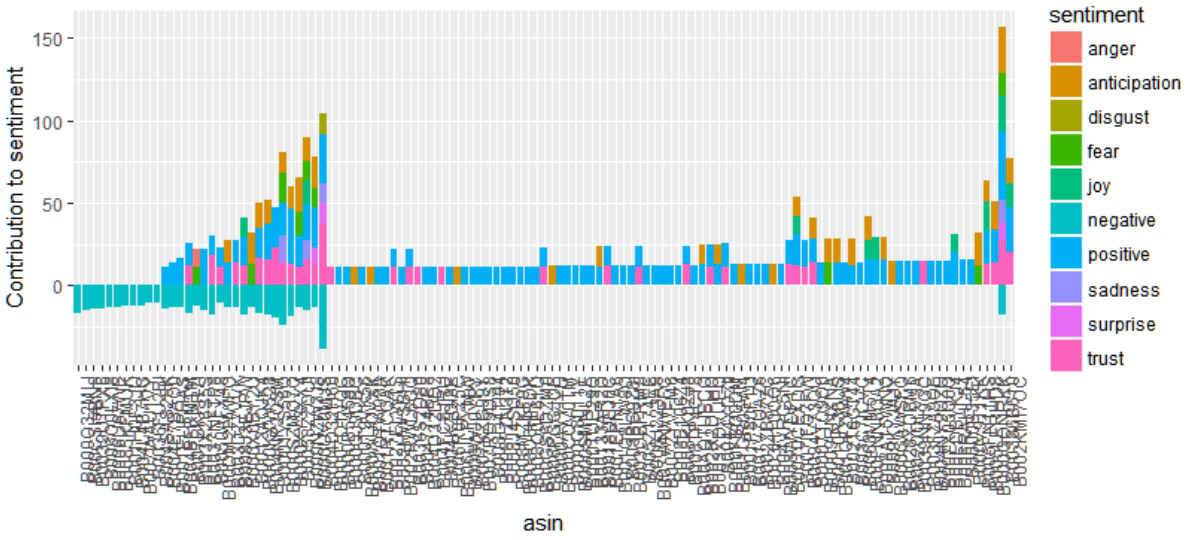
Conlusion :

Dans ce projet R nous avons appris à utiliser un système de recommandation basé sur du sentiment analysis de review d'utilisateur avec un schéma de construction de type UCBF.

François d'Anselme (faolin sur Git)
Grégoire de Charnace(gregoireDC sur Git)
Philippe Kreiss



Annexe 1 : visualisation de bing



annexe 2 : visualisation de nrc