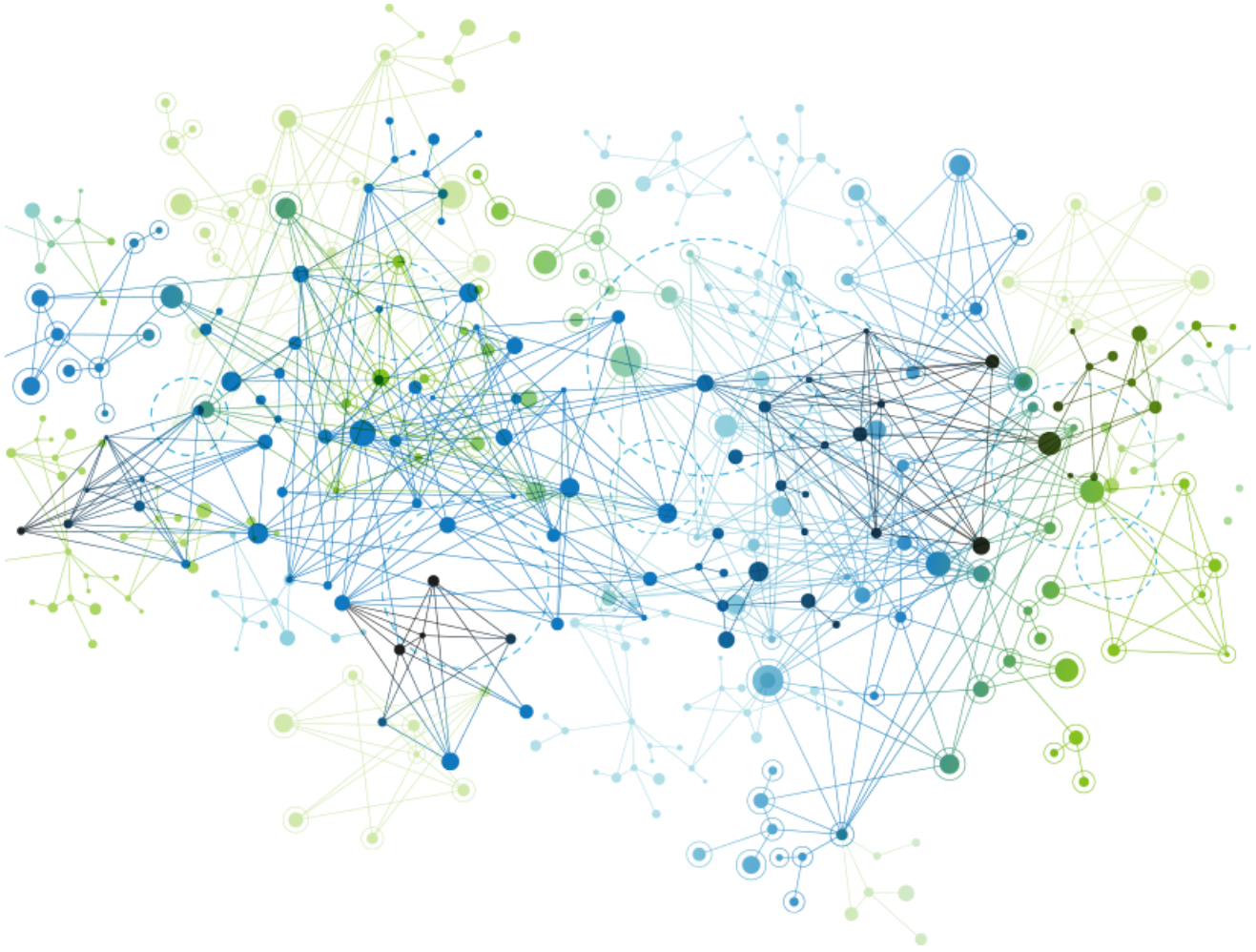
POLYTECH LILLE | Université de Lille

Rapport de projet Big Data



IS4

2022/2023

Introduction

Dans le cadre du projet text mining en big data il nous a été demandé de déployer des modèles de machine learning afin de prédire des commentaires toxiques.
Afin d'analyser les commentaires toxiques nous avons tout d'abord nettoyé notre base de données en éliminant les mots sans importances comme les conjonctions de coordinations "mais", "où", "et", "donc" et autres déterminants non pertinents pour notre étude puis d'autres opérations de vectorisation ont été fait par la suite. Notre colonne à la base de toutes nos analyses sera la colonne "comment_text". Plusieurs approches ont été utilisés pour différents modèles

1)Les différentes approches

1.1 Random Forest

Avec le Random forest j'ai tenté plusieurs approches. Avec un countvectorizer, de 200 mots, de 150 puis 300 mots . Après avoir augmenté le nombre de mots j'ai eu aussi à augmenter la profondeur de mes arbres . J'ai donc eu de meilleures performances grâce à cela. Puis j'ai tenté de labéliser les lignes en fonction de si elles contenaient le mot fuck. J'ai donc donné un poids plus important aux lignes contenant ces mots. Ensuite j'ai tenté d'autres vectorisation comme l'IDF et le word2vec et à chaque fois j'ai étudié les faux négatifs et les vrais positifs.

Tableau des résultats avec un vocabulaire de 300 mots

Radomn Forest	fmeasurebylabel	vrai positif	faux négatifs	faux positf
IDF	90,22	153	125	409
COUNTVECTORIZER	83,76	181	381	7

Chacune des méthodes a ses avantages. Globalement on pourrait dire que le IDF est meilleur grâce fmeasurebylabel. Pourtant on remarque que le taux de faux positifs est plus grand . Il est aussi meilleur en termes de faux négatifs. Toutefois il n'est pas bon en termes de faux positifs .

Tableau des résultats avec un vocabulaire de 300 mots avec la labellisation

Radomn Forest	fmeasurebylabel	vrai positif	faux négatifs	faux positifs
IDF	89,29	562	560	2334
COUNTVECTORIZER	92,56	534	0	2334
W2VEC	33,03	76	486	16

Après avoir labellisé on remarque qu'il n'y a plus de faux négatifs avec le countvectorizer mais beaucoup de faux positifs. Le modèle avec le plus petit nombre de faux

positifs est celui vectorisé par W2VEC. Il faudra faire un compromis entre le plus petit nombre de faux positifs et négatifs et dans ce cas là ce serait le W2VEC le meilleur malgré le faible nombre de vrai positif.

Je pense que ce n'était peut être pas une si bonne idée de mettre un poids cinq fois plus grand aux lignes contenant toxiques , peut être que j'aurais dû en mettre moins pour les vectorisations de type IDF et COUNTVECTORIZER.

1.2 Decision Tree

Tableau des résultats avec un vocabulaire de 300 mots

arbre de décision	fmeasurelabel	vrai positif	faux négatifs	faux positifs
IDF	90,23	76	486	16
COUNTVECTORIZER	91,91	195	367	22

En utilisant le même processus qu'en partie précédente j'ai obtenu de meilleurs résultats avec le countvectorizer. Avec cette méthode, le modèle prédit mieux les commentaires toxiques. Les faux négatifs sont moins nombreux qu'avec le countvectorizer .

Tableau des résultats avec un vocabulaire de 300 mots avec la labellisation

arbre de décision	fmeasurelabelv	vrais positifs	faux négatifs	faux positifs
COUNTVECTORIZER		562	0	2334
IDF		562	0	2334

Ici le l'indicateur n'a pas pu être calculé. Seuls les indicateurs de la matrice de confusion ont été calculés.
J'obtiens les mêmes résultats avec les deux ce que je trouve assez bizarre.

Conclusion

En conclusion je me suis basé sur 2 types de modèles : les random forest et les arbres de décisions. Les résultats sont plutôt concluants sauf concernant la labellisation dans le cas de l'arbre de décision. J'aurais voulu essayer d'autres modèles comme la régression logistique et donné des poids différents à mes observations selon d'autres critères que la contenance du mot "fuck" .