SQY: Résultats de recherche

Analyser l'état d'un réseau de transport avec du NLP

Rétrospective de la problématique

Saint-Quentin-en-Yvelines (*SQY*) (78180) est une ville possédant un fort réseau de transport en commun. Ces transports peuvent être susceptible à des incidents, qui mal interprétées peuvent bloquer toute une ligne de transport.

L'objectif est donc d'analyser les infos trafic publiées sur *Twitter* par les *Community Manager*[1] (*CM*) des différents réseaux de transport, et ceux dans le but d'informer les usagers des retards/annulations de leurs transports depuis l'application *Viago*.

Pour cela, SQY à fait appel à l'*Atelier* d'*EPITECH* (Paris) pour qu'il effectue une analyse de faisabilité et un étude de l'existant pouvant répondre à la problématique énoncée.

Visualisation de la donnée

La première étape de l'étude consista à analyser les différents tweets postés par les CM les 15 derniers jours résultant en un total de 200 tweets différents. La méthode utilisée a été de constituer un *nuage de mots*[2] à l'aide d'un programme informatique (*script*).

La première analyse s'est effectuée sur la donnée inchangée, l'ensemble des mots[3] utilisés dans les différents tweets ont été pris en compte.

[1] Comptes utilisés: *RERC_SNCF*, *lignesNetU_SNCF*, *RemiTrainPCLM*, *InfoSqybus*, *CarsHourtoule*, *StavoInfoLignes*, *TransdevSud78*, *ALBATRANS91*, *Actu_Savac*

[2] Représentation visuelle des mots-clés les plus utilisés

[3] Il est important de noter qu'un mot possédant une majuscule ou un accent est considéré comme différent de celui qui n'en a pas. (exemple: "*Mot*" est différent de "*mot*")

Résultat de la première analyse de donnée

Le résultat contient de nombreuses prépositions et pronoms qui empêchent une bonne visualisation de la récurrence des mots clefs.

La deuxième analyse s'est effectuée après avoir retiré les mots appelés "*stop words*[4]" et retiré toutes les majuscules.

Résultat de la deuxième analyse de donnée

On remarque que les tweets ne suivent pas une syntaxe précise, il est donc difficile de détecter un paterne dans la donnée uniquement avec un nuage de mots.

La suite de la visualisation de la donnée s'est effectuée avec l'outil Doccano un outil de labellisation.

Un total de 200 tweets ont été explorés et labellisés pour visualiser le format de la donnée et tenté d'identifier des parternes.

Malheureusement la donnée se trouve être disparate.

[4] Mot courant qui n'apporte pas de context à la donnée

Quelques exemples de tweet:

Exemple n°1:

@papillons007 @train_nomad @SNCFTERAURA @TERPays2LaLoire @TER_BFC_Trafic @TERBreizhGo @TERHDF @TERGrandEst @TERNouvelleAQ @TERSUD_SNCF @TER_Occitanie @TERCentreTrafic @RemiTrain Merci ! Bon après-midi à vous également

Exemple n°2:

@SegoSaulnier @SegoSaulnier Bonjour, selon mes informations, votre prochain départ est prévu à 9h20 (Date) . Notez que le trafic est fortement perturbé (Context) sur toute la ligne. Je reste disponible.

Exemple n°3:

RT @IDFmobilites: Avec l'application Île-de-France Mobilités, il est maintenant possible de recharger son passe #Navigo directement avec…

Exemple n°4:

❌ #InfoTrafic # RERC (Ligne)

Le prochain train VICK (Ligne) en gare de Juvisy (Arret) est prévu à 10h31 (Date) , arrivée à Versailles Château (Arret) à 11h34 (Date) .

Exemple n°5:

@descartes59000 @descartes59000 Bonjour, les trains sont directs entre de Juvisy (Arret) vers Brétigny (Arret) ce week-end. Des travaux sont effectués sur les voies. Des bus de remplacement sont mis en place. Retrouvez toutes les infos ici >> https://t.co/Gi8KQS3gFl

Malgré une répétition de certains formats (exemple n°4), il n'y a pas de réels paternes récurrents dans la donnée que ce soit sur le contenu du tweet ou l'ordre d'écriture des informations.

De simple technique de parsing[5] de la donnée ne suffiront pas, il va falloir faire usage de technique avancée de NLP[6] pour tenter de maximiser les résultats.

[5] Récupérer des mots dans une phrase en suivant une suite logique.
[6] Natural Language Processing

Exploration et Benchmark des techniques de NLP

La suite de l'étude s'est portée sur l'exploration et le benchmarking[7] des différentes solutions de NLP, parmi lesquelles nous en avons retenu deux:

- FLAIR via NER[8] sans fine-tuning[9]
- Hugging Faces via CamemBERT[10] NER.

Les résultats étaient cohérents mais encore trop imprécis pour être utilisés en production.

Exemple de phrase fournie au modèle:

```
PERTURBATION INOPINEE LIGNE 43


En raison d'une fuite d'eau sur le Chemin de la Ratelle à Fontenay-le Fleury,
nous vous informons que les arrêts entre Victor Hugo et Paul Eluard inclus, ne
seront pas desservis jusqu'à la fin de l'intervention des secours.
```

Résultat avec FLAIR

```
[<MISC-span (2): "INOPINEE">, <LOC-span (12,13,14,15): "Chemin de la Ratelle">,
<LOC-span (17,18): "Fontenay-le Fleury">, <PER-span (27,28): "Victor Hugo">,
<PER-span (30,31): "Paul Eluard">]
```

Il serait donc nécessaire d'adapter un modèle (fine-tuning) au contexte du projet.

Méthode conseillé

Pour résoudre au mieux les enjeux du projet, nous proposons deux solutions:

CamemBERT

La première possibilité est d'utiliser un modèle pré-entraîné sur un corpus de texte français et le fine-tuner. Un modèle qui pourrait correspondre est *CamemBERT* qui est disponible sur ce lien[11].

Pour cela il faudrait procéder en deux étapes:

Labellisation

La première étape consiste à labelliser la donnée, la quantité nécessaire à labelliser n'a pas été estimée mais ça serait de l'ordre de plusieurs centaines de tweets.

À noter que la labellisation est une tâche longue et qui doit être faite rigoureusement.

---

[7] N'a été pris en compte que les résultats des prédictions et non les délais/coûts de prédiction
[8] Named Entity Recognition
[9] Réentraîner un modèle pré-entraîné
[10] Adaptation de BERT sur un corpus français
[11] camembert-model.fr

Entrainement

Une fois un jeu de données suffisant, il faut entraîner le modèle à prédire les labels qui ont été précédemment créés.

Une fois le programme informatique créé, cette tâche ne demande pas beaucoup de supervision mais reste chronophage et demande beaucoup de puissance de calcul.

Conclusion de l'approche

Cette approche est chronophage et demanderait des coûts de développement notable pour obtenir des résultats incertains.

De plus, l'une des difficultés rencontrées à été de déduire l'information à transmettre depuis les prédictions du modèles, c'est pourquoi ce n'est pas la méthode que l'on conseille.

Formater la donnée

La deuxième méthode et celle que nous recommandons est de formater la donnée écrite dans les tweets.

Si la donnée prend toujours la même forme, il est simple d'automatiser la récupération d'information.

Pour rendre la donnée lisible par les utilisateurs et interprétable par un programme informatique, il est possible de mettre en place un formateur de données qui serait utilisé par les CM.

Les CM auraient alors juste à indiquer les informations liées au trafic (ligne, date, type d'incident) et le formateur rédigerait le tweet.

Conclusion de l'approche

Cette approche serait moins coûteuse, plus rapide et plus efficace. Toutefois, cela nécessite que les CM s'accordent à utiliser un format de données et ne pas le modifier sous peine de devoir changer le programme informatique.