Rapport du projet de text mining : Création d'un agent de dialogue

DOUX Julie, GUILLARD Marion, LENOBLE Inès

Ci-dessous se trouve le lien vers le répertoire GitHub contenant nos différents fichiers et nos données :

https://github.com/j-dx/projet_text_mining

- ## Présentation de l'architecture et du fonctionnement

Notre bot permet de communiquer avec un agent conversationnel sur le sujet des voyages. Nous avons en effet scrappé les FAQ des sites Tui et Exotismes, dont les liens sont : https://www.tui.fr/faq/ et https://www.exotismes.fr/voyages/t/FAQ.html. Ces sites proposent des voyages comprenant des vols et des hôtels.

Afin de commencer à échanger avec le bot, il faut télécharger l'ensemble du dossier Github "projet_text_mining". Le sous répertoire "aide_creation" n'est pas indispensable au bon fonctionnement, il montre les démarches que nous avons suivies pour mettre en place le bot.

Le bot débute en lançant le fichier python "code_final" . Il sera peut-être nécessaire de télécharger le modèle spacy "fr_core_news_sm" à l'aide de la commande "!python -m spacy download fr_core_news_sm".

Après réception de la demande, le bot prétraite la question et détermine si elle appartient ou non à la thématique du voyage. Si la demande n'est pas en lien avec le thème, il génère une réponse aléatoire originale. Sinon, il effectue un nouveau prétraitement sur la question initiale et cherche la question ou la réponse la plus similaire à la demande et renvoie la réponse adaptée.

- ## Mise en oeuvre

Prétraitement et classifieur binaire

Un premier modèle binaire permet de distinguer les thématiques liées au métier des autres thématiques. Nous avons donc utilisé les données des deux FAQ (questions et réponses) et les données d'une partie du corpus de la base conversationnelle. Nous avons appliqué un traitement au jeu de données avant de lancer le modèle. En effet, nous avons commencé par lemmatiser les données. Ce prétraitement est suivi d'une tokenization.
Le meilleur modèle est la SVM (Support Vector Machine) en utilisant un vectoriseur binaire spécifiant les mots vides. De ce modèle, nous récupérons la probabilité d'appartenir à la classe thématique, si celle-ci est inférieure à 0.15, nous supposons que la demande est hors-sujet. En effet, nous avons décidé de ne pas supposer que la probabilité que la personne parle du thème soit d'une chance sur deux. Etant donné que le bot a été conçu

pour répondre aux demandes concernant les sites de voyage, il y a de fortes chances qu'elle souhaite dialoguer sur son futur voyage.

Génération de texte

Le corpus que nous avons utilisé correspond à un corpus de sous-titres de films en français trouvé sur le site suivant : http://opus.nlpl.eu/OpenSubtitles-v2018.php . Afin d'exploiter celui-ci, nous nous sommes inspiré de ce site : https://www.tensorflow.org/tutorials/text/text_generation pour apprendre notre modèle. Nous avons commencé par utiliser un modèle basé sur des mots, puis nous avons décidé de garder le modèle basé sur les caractères car il était plus performant pour générer du texte. Cependant, nous n'avons pas pu prendre l'ensemble du corpus à cause de la capacité de nos machines. De plus, nous avons seulement entraîner le modèle sur une époque et sur un huitième du corpus, car le noyau python se déconnectait systématiquement.

Prétraitement et matrice de similarité

Lorsque la demande est jugée dans le thème du voyage, celle-ci est prétraitée. Une lemmatisation est appliquée. De plus, nous remplaçons les URL et les noms des deux agences par des expressions générales. Ce prétraitement est suivi d'une tokenization. Après cela, nous utilisons la matrice de similarité pour trouver la question ou la réponse des deux FAQ la plus similaire à la question de l'utilisateur. Le bot renvoie la réponse adéquate.

- **Procédure d'évaluation et résultats des évaluations**

Évaluations quantitatives

Pour le classifieur binaire, nous avons obtenu sur un jeu de données test, un taux de bonne classification de 0.96.
Pour évaluer si le bot répondait avec la bonne question, nous avons créé un jeu test. Nous avons obtenu un taux de bonne classification de 0.32 sur ce jeu. Cependant, certaines explications peuvent être apportées pour justifier ce résultat, nous les évoquerons dans la partie évaluations qualitatives.

Évaluations qualitatives

Les réponses sont parfois justes mais comptées comme fausses. En effet, nous avons scrappé deux sites de voyages différents et de nombreuses questions se ressemblent. Par exemple, pour la question "Quelles assurances conseillez-vous de prendre ?", nous acceptons ces deux réponses :

- Aucune assurance ou assistance n'est incluse dans nos voyages. En association avec Présence Assistance, nous vous proposons plusieurs types d'assurance. Retrouvez ici toutes les informations sur les assurances https://www.tui.fr/pratique/assurances-voyage/
- Exotismes propose une seule assurance voyage qui est l'assurance MULTIRISQUES. Pour plus de renseignements, cliquez sur le lien ci-dessous: https://www.exotismes.fr/documents/assurances/

Pour les autres phrases qui ne correspondent pas, nous avons l'impression que les mots importants tel que "adolescents" n'ont pas les poids les plus grands. On pourrait croire que les poids sont mal répartis sur les mots d'une phrase. Il faudrait peut-être ajouter des mots vides ou effectuer un autre prétraitement. Afin d'améliorer ce résultat, nous avons testé plusieurs prétraitements avec vectoriseurs et matrices de similarité différents. Cependant, nous n'avons jamais réussi à améliorer le score. Le meilleur résultat a été obtenu avec une matrice de similarité binaire et un grammage allant de 1 à 2.

Malgré le fait que nous avons augmenté la probabilité d'appartenir à la classe "thématique" comme évoqué précédemment, des problèmes de classification sont encore présents.

De plus, il suffit parfois de peu pour être bien classé, si nous demandons "Les animaux sont-ils acceptés ?" du texte est généré. En revanche, si nous écrivons "Les animaux de compagnie sont-ils acceptés ?", il répond la bonne réponse.

- **<u>Ressenti personnel</u>**

Limites et points forts du bot

Le bot détermine plutôt bien si la demande est en lien avec le sujet. Cependant, la matrice de similarité ne renvoie pas de très bon résultat. De plus, le fait que le générateur de texte n'a pu être entrainé suffisamment longtemps, le texte généré est plutôt incohérent.

Difficultés rencontrées

Lors de ce projet, nous avons été confrontées à certaines difficultés. L'une d'entre elles était de choisir la taille du corpus que nous devions garder pour générer du texte. En effet, il a été difficile de trouver la taille à conserver. Lorsque nous prenions tout le fichier, la base était trop volumineuse pour lancer le modèle. A l'inverse, quand nous diminuons la taille de la base, le modèle était peu performant et le texte généré n'était pas cohérent.

Un autre problème que nous avons rencontré est le fait que beaucoup de questions se ressemblent. Effectivement, comme nous avons scrappé deux sites différents basés sur le

même thème, il y a donc certaines fois plusieurs réponses à la même question. Ceci pose notamment problème pour la création du jeu test.

Enfin, nous avons essayé d'utiliser un modèle multi-classe permettant de distinguer les catégories. En effet, les questions sont rangées dans différentes catégories dans les deux FAQ scrappées. Nous avons donc utilisé ces données classées en catégories. Le meilleur classifieur se basait sur une forêt aléatoire en utilisant un vectoriseur binaire spécifiant les mots vides. Ce meilleur modèle possédait un score de 0.5 (accuracy), donc nous avons décidé de ne pas l'utiliser pour choisir la catégorie. Nous pensons que le classifieur était mauvais car il n'y avait pas assez de questions et de réponses dans chaque catégorie. Nous avons pu remarquer que certaines catégories avaient seulement deux questions.

Améliorations

Le prétraitement pourrait être développé afin d'améliorer le score du jeu de test car il est petit. Un enrichissement de la base serait aussi sûrement bénéfique.

Un autre point serait d'augmenter la taille du corpus utilisé pour la génération de texte ainsi que le temps d'entraînement.

Une évaluation extérieure aurait pu nous aider à connaître des défauts que nous n'avons pas pu voir. Néanmoins, nous n'avons pas eu assez de candidats sur le temps qui nous était imparti.