PLATEFORMES LOGICIELLES POUR LE TAL

RAPPORT - EXERCICE SIMILARITÉS II

Morgann Sabatier

Master 2 - Langue et Informatique

23/01/2022

1 Introduction

Dans le cadre de ce travail, nous avons travaillé sur un ensemble de données. Il s'agit de 1000 questions, 1000 réponses attribuées ainsi que pour chaque couple question-réponse, un identifiant. L'objectif était de déterminer la distance minimale. Pour ce faire, nous avons procéder en plusieurs étape.

2 Données

Tout d'abord, il était nécessaire de prendre en compte le format des données (.tsv), nous avons choisi d'utiliser la bibliothèque pandas et d'ajouter u header au DataFrame pandas, ce qui nous permet d'organiser au mieux nos données tout en accédant aux colonnes que nous désirions.

Nous avons également choisi d'observer nos données en regardant la longueur de chaque question et de chaque réponse que nous avons présenter sous forme d'histogrammes. Nous nous sommes alors rendu compte que la longueur des réponses était bien plus hétérogènes que les pour les questions. Etant donné le calcul de similarité, il est envisageable que les réponses extrêmement longues ne seront pas associées à leur réponse, tandis que les réponses très simples et courtes seront attribuées à de multiple questions.

3 Cheminement

Ensuite, nous souhaitions mettre en place une pipeline permettant de vectorizer l'ensemble des questions et réponses. Il nous a semblé évident que nous ne souhaitions pas comparer toutes les questions à toutes les questions et toutes les réponses (c'est coûteux et nous souhaitons obtenir le couple question-réponse la plus proche). Nous avons donc choisi de vectorizer chaque question et toutes les réponses afin d'obtenir notre matrice de distance.

A partir de cette matrice, nous avons sélectionné la valeur la plus basse de la première ligne.

Tout ceci a été effectué pour plusieurs types de calcul de distance.

Nous avons conservé un ensemble de résultat sous format json. Les résulats sont attachées à l'archive jointe du rendu.

Enfin, nous avons observer que les résultats n'étaient pas très élevés (52% d'exactitude), nous nous sommes donc posé la question de savoir pourquoi.

Tout d'abord, une faible proportion des réponses est sélectionnée. Nous avons donc choisi d'observer ces réponses pour nous rendre compte qu'il s'agit simplement de réponse très courte incluant des mots interrogatifs et la ponctuation des questions. On se retrouve donc avec des phrases qui sont extrêmement similaires aux questions de la première colonne, d'autant plus qu'on a pu observer que les questions sont souvent très courtes.

4 Idées d'amélioration

Tout d'abord, la manière dont nous avons procédé pour coder notre pipeline pourrait être optimisée, notamment le calcul du minimum. Quelques optimisations du temps de calcul de la pipeline pourraient se faire.

Ensuite, nous avons pensé à plusieurs solutions pour améliorer les résultats de similarité. Tout d'abord, la suppression des mots interrogatifs dans les réponses (ou les deux). Ensuite, nous avons pensé à supprimer les ponctuations et enlever les réponses trop disproportionnées qui ne seront jamais détectées avec cette méthode.

Enfin, peut-être qu'une méthode d'apprentissage non supervisée pourrait aider à une première détection, notamment pour une détection de thèmes. Ensuite, au sein des clusters, nous pourrions tenter d'associer question et réponse.

Cet exercice nous démontre qu'une tâche aussi simple qu'attribuer automatiquement une question à une réponse s'avère bien plus complexe que nous le pensions. Il nous a permis de nous concentrer sur la segmentation de notre pipeline et de prendre du recul sur nos résultats et notre a priori de la tâche.