Documents structurés TALA540A

Évaluation des modèles Spacy en chinois sur les OOV sur des corpus hors domaine

Anonymous submission

Abstract

Partant d'un corpus organisé en domaines ou genres textuels, on choisit d'évaluer l'exactitude de l'étiquetage des POS des OOV du modèle Spacy lorsqu'il est entraîné sur plusieurs domaines et testé sur un corpus hors domaine.

1. Les données

Le Lancaster Corpus of Mandarin Chinese (LCMC) a été construit sur le modèle des corpus d'anglais et d'américain modernes FLOB et FROWN. Il est organisé en 15 domaines ou genres textuels variés qui vont des textes littéraires au textes académiques, en passant par des textes humoristiques et des dépêches. Tous ces textes ont été publiés en RPC au début des années 1990. Le LCMC est composé de deux répertoires contenant chacun 15 fichiers au format xml. Un répertoire contient les fichiers avec les phrases en caractères chinois et l'autre répertoire contient les fichiers avec les phrases en pinyin et les tons. On s'intéresse au répertoire qui contient les phrases en caractères chinois. Par ailleurs, le jeu d'étiquettes POS utilisé par Lancaster comprend 50 étiquettes. Or, avec le modèle Spacy nous allons utiliser l'étiquetage UD qui en comprend 15. Il nous faudra donc réétiqueter le corpus avec Spacy.

Réorganisation des sous-corpus Après avoir parcouru les textes on cherche à regrouper les domaines par affinité de genre textuel, on s'appuie en fait sur les titres des domaines déjà identifiés. L'objectif est de conserver un découpage par domaine mais d'avoir plus de données par domaine et moins de sous-parties. Tout cela reste très expérimental et intuitif. Finalement, on a :

- Le sous-corpus News qui est le regroupement des fichiers News reportage, News editorials News reviews. Il représente 18% du corpus total en nombre de textes et sa taille est de 5,3 Mo.
- Le sous-corpus Populaire qui regroupe les fichiers de corpus Skills, trades and hobbies, Popular lore, et Humour. Il représente 18% du corpus total et sa taille est de 5,4 Mo.
- Le sous-corpus Fiction qui regroupe les fichiers de General Mystery, polar, SF, aventure militaire et romantique. Il représente 23% du corpus total en nombre de textes et sa taille est de 5,4 Mo.
- Les sous-corpus Religion (4%, 1 Mo), Essais (15%, 4,6 Mo), Divers rapport (6%, 1,9 Mo) et Sciences académiques (16%, 4,9 Mo) semblant de genre homogènes, n'ont pas été regroupés. Ils présentent un grand écart de tailles.

Les corpus comparables en tailles sont autour de 5 Mo et représentent les domaines : news, populaire, essais, science académique et fiction.

2. Que va-t-on évaluer à partir de ces sous-corpus?

On veut exploiter ce découpage en genres pour évaluer au moins deux modèles de Spacy. L'objectif est de voir à quel point, après avoir été entraîné sur un corpus contenant tous les domaines sauf un, le modèle arrive à étiqueter les mots nouveaux du domaine qu'on lui donne à étiqueter. On va regarder les scores d'exactitude obtenus pour l'étiquetage POS des OOV.

3. Les modèles

Pour le chinois Spacy propose 4 modèles. Nous allons entraîner sur les deux suivants:

- zh_core_web_mdb, basé sur tok2vec. La précision annoncée pour les POS (fine grained tags, Token.tag) est de 0.90.
- zh_core_web_trf basé sur un transformer (Bert). La précision annoncée pour les POS (fine grained tags, Token.tag) est de 0.92.

4. La procédure

Script On reprend le script `eval_sous_corpus.py` vu en TP, mais on doit l'adapter. Pour préparer les données on ajoute la fonction `xml\_to\_conll()` qui extrait au moyen de regex, les métadonnées identificateur du genre textuel et les tokens dans les corpus .xml. On ne récupère pas les POS puisqu'on ne va pas pouvoir les exploiter.

Le découpage des sous-corpus par domaines étiquetés avec Spacy en train et dev ont été fait manuellement (bash head -n et tail -n >). On prend 10% du fichier (nbre de lignes) pour créer le fichier dev.
Les sous-corpus sont concaténés pour obtenir 7 corpus train et 7 corpus dev dont les noms de fichier sont au nom du domaine manquant, par exemple le sous_corpus_train_sci_aca.conllu contient les données de tous les domaines sauf celle du domaine sciences académiques.

Les sous-corpus train et dev

- sous-corpus populaire: 'divers','news','essais','fiction','religion','sci_aca')
- sous-corpus sci_aca: ('divers','news','essais','fiction','religion','populaire')
- sous-corpus religion: ('divers','news','essais','fiction','sci_aca','populaire')
- sous-corpus fiction: ('divers','news', 'essais','religion','sci_aca','populaire')
- sous-corpus essais: ('divers','news','fiction','religion','sci_aca','populaire')
- sous-corpus news: ('divers','essais','fiction','religion','sci_aca','populaire')
- sous-corpus divers: ('news', 'essais','fiction','religion','sci_aca','populaire')

Conversion des corpus au format spacy Pour la conversion au format .spacy le fichier .conllu doit bien présenter une ligne vide à chaque fin de phrase et notamment en fin de fichier et au début. C'est une norme à connaître quand on concatène nos fichiers .conllu.

Entraînement À chaque entraînement, le vocabulaire extrait est celui du corpus train donc de tous les sous-corpus excepté le corpus qui est testé (corpus test).
La quantité de données est assez importante et l'entraînement du modèle a été long et suivait une courbe qui progressait régulièrement (passage à la dizaine supérieure) à chaque changement d'epoch.

5. Résultats avec le modèle zh_core_web_mdb

Les résultats des tags de oov sont très bas, mais je n'ai pas eu le temps de faire des essais en modifiant les paramètres (batch_size, max_steps, max_epochs).

domaine	OOV	f1
divers	0,38	0.70
essais	0,17	0,31
fiction	0,49	0,75
news	0.43	0,65
populaire	0,52	0,72
religion	0,48	0,71
sci_aca	0,53	0,71

Table 1: Résultats OOV : tags corrects des oov sur le nombre total de oov.

Les 5 domaines de tailles similaires autour de 5 Mo: 'news', 'populaire', 'essais', 'sci_aca', 'fiction' ne présentent pas forcément des résultats proches. Par exemple, le domaine "divers" a une taille de 1,9 Mo mais un résultat d'oov correctement taggés meilleur que celui du corpus du domaine "essais" qui est beaucoup plus gros.

6. Conclusion

La préparation des données a pris beaucoup de temps. Il a fallut notamment mettre au point la fonction de conversion mais également réfléchir à la meilleure manière de regrouper les sous-corpus au niveau du script tout en les traitant. D'autre part, je n'ai pas eu assez de temps pour bien comprendre ces résultats et à la lumière de ces derniers, de modifier les paramètres du modèle et peut-être même de de revenir sur la l'examen des données ou la chaîne de traitement. Par contre, ce travail m'a permis de manipuler les données de procéder à des découpages de corpus, de préparer l'entraînement et d'entraîner un modèle.