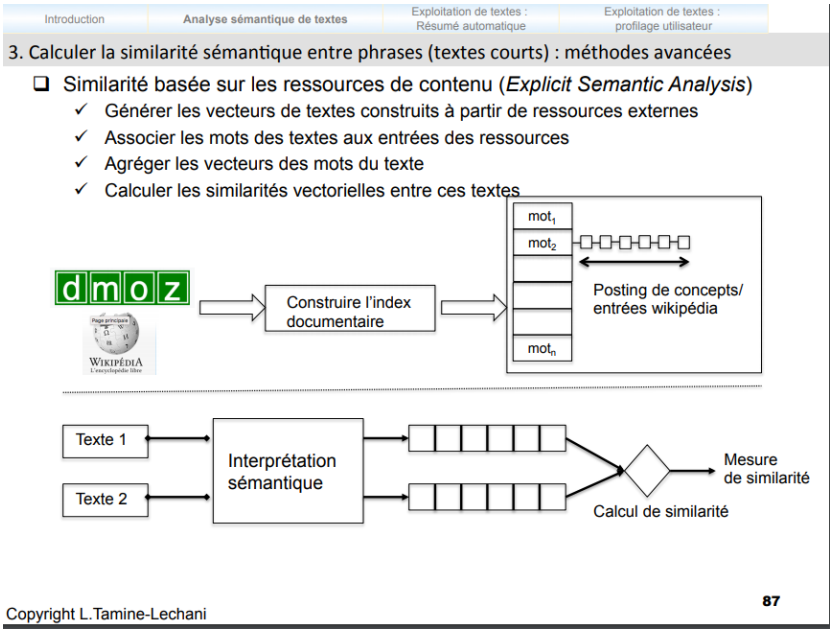
Université Paul Sabatier
EIMAB3H1 - Analyse et exploitation de données

Enseignant : José G. Moreno

30/09/2019

TP 2. Similarité sémantique multi-niveau

Rappelez vous du calcule de l'ESA (Explicit Semantic Analysis). Notez que pour appliquer cette méthode il faut d'abord construire de vecteurs pour représenter chaque mot.



Dans le TP1, nous avons construit de vecteurs pour chaque mot en utilisant la fenêtre de contexte et la PPMI et aussi utilisé les vecteurs appelés *word embeddings*. Dans ce TP, les vecteurs seront construits en utilisant la fréquence d'un mot dans un document.

1. Téléchargez le fichier simplewiki et simplewiki.1000.lines sur moodle.

2. Pour chaque mot, calculez sa fréquence par rapport à toutes les pages Wikipédia. Pour le faire, construisez une matrice avec autant de colonnes comme de pages Wikipédia et autant de lignes comme de mots dans le vocabulaire. Chaque position ***X_ij*** contiendra la fréquence du mot ***i*** dans le document ***j***. Vous pouvez simplifier la tâche, par exemple, en utilisant uniquement que les documents dans les premières 1000 lignes de la Wikipédia.

3. Utilisez la formule d'ESA pour calculer la similarité entre le mot "leptodactylid" et les mots "frog", "fish", "bird" and "dog". Vos résultats sont-ils cohérents?

4. Utilisez les vecteurs du TP1 (*word embeddings*) et recalculez les similarités précédentes.

5. Notez qu'un seul exemple ne nous permet pas de trouver la meilleur méthode/représentation car le résultat peut corresponde à une donnée aberrante. Pour faire une évaluation plus juste, vous pouvez utiliser des collections standards dédiées à ce type de tâches. Par exemple, la collection SemEval2014task3 contient de pairs de mots et phrases pour lesquelles il faut trouver une valeur de similarité automatiquement. Dans cette collection, la phrase "loss of air pressure in a tire" doit être comparée avec la liste de mots "flat-tire", "puncture", "tire", "parking" et "butterfly", et votre système doit décider un dégré de similarité pour chaque mot. Une évaluation de differentes méthodes est possible car pour chaque mot une valeur de similarité a été donné par un humain. Votre tâche consiste en automatiquement trouver les valeurs de similarité qui correspondent le mieux à une prédiction humaine. Pour le faire :

- Téléchargez le fichier semeval2014task3PhraseWord.zip disponible sur moodle.
- Lisez le fichier test/phrase2word.train.input.tsv est calculez la similarité entre chaque pair de phrase et mot avec les méthodes ESA, Wordnet et *word embeddings.*
- Sauvegardez la similarité pour chaque pair dans un nouveau fichier et appelez-le *output.txt*. Chaque valeur doit être dans une nouvelle ligne, donc le nombre de lignes dans le fichier test/phrase2word.train.input.tsv et le fichier que vous avez crée doit être le même.
- Évaluez la performance en utilisant la commande

```
java -jar evaluation/task-3-scorer.jar test/phrase2word.test.gs.tsv
output.txt
```

  - Vous avez un fichier de test pour l'évaluation dans de dossier baselines. Si vous utilisez la commande

```
java -jar evaluation/task-3-scorer.jar test/phrase2word.test.gs.tsv
baselines/phrase2word.test.baseline.tsv
```

  - Vous devez avoir comme résultat

```
Scores for SemEval-2014 Task 3
Pearson's correlation   0.164760    Spearman's rho    0.161558
```

- Renseignez-vous sur les deux métriques utilisées.
- Évaluez chaque méthode de représentation sémantique (ESA, Wordnet ou *word embeddings)*. Quelle méthode donne les meilleurs résultats ?