NATURAL LANGUAGE PROCESSING (NLP)

Quelques concepts avancés

Abdoul Kader KABORE
abdoulkader.kabore@protonmail.com

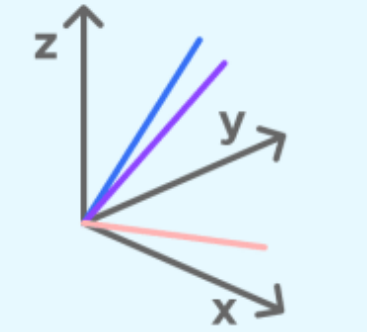
PLAN

1. Introduction au NLP
2. Autres concepts avancés
    1. Embeddings
    2. N-Grams
    3. Embedding de phrasess
3. Travaux Pratique

Newspaper ⟨0.08, 0.31, 0.41⟩

Magazine ⟨0.09, 0.35, 0.36⟩

Biking ⟨0.09, 0.35, 0.36⟩



Newspaper

Magazine

Biking

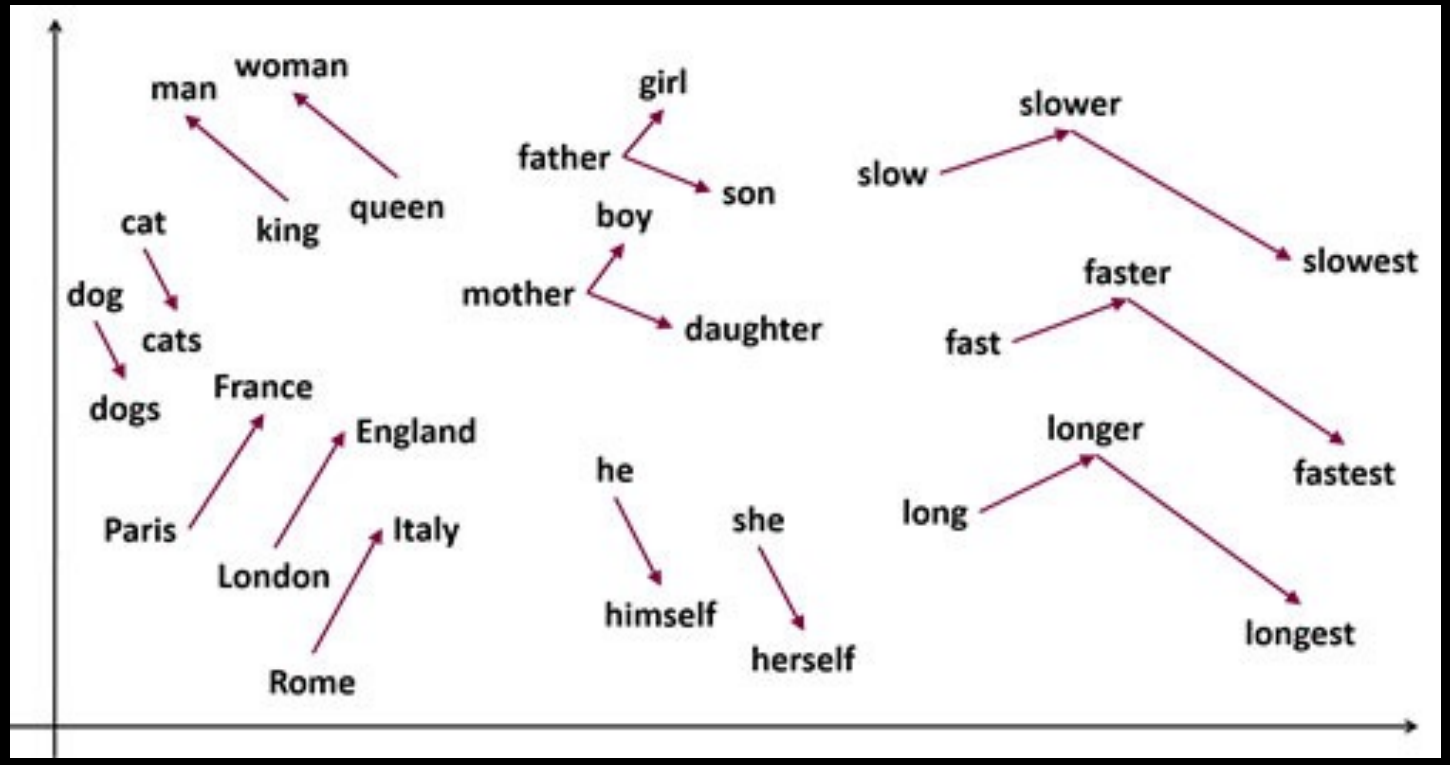
WORD EMBEDDINGS

Le principe des word embeddings (nommés parfois en français « plongement lexicaux ») est de représenter l'ensemble des mots d'un « dictionnaire » sous la forme de vecteurs à valeurs numériques réelles.

L'objectif est que les représentations vectorielles des mots soient proches lorsque ces mots correspondent à des contextes similaires

WORD EMBEDDINGS

- Exemple



WORD EMBEDDINGS

- Exemple avec spacy

```
[12]: import spacy

      nlp = spacy.load("fr_core_news_sm")

      doc = nlp("Bonjour le monde")
      doc[0].vector
```

```
[12]: array([ 1.5168922 ,  1.8968501 , -5.266554  ,  0.5778411 ,  3.1850119 ,
              1.7140923 , -2.7160392 , -0.4749193 ,  0.75266796, -0.3379007 ,
              0.93404543, -0.3991629 ,  1.2780753 , -2.3791323 ,  2.2031436 ,
             -0.48126447,  3.411249  ,  1.4551203 , -2.9011712 , -2.5047753 ,
             -1.1931838 , -3.7717817 , -1.2136084 ,  2.9315577 , -0.49599808,
             -2.963449  , -0.36337614,  3.1553857 ,  6.2001705 , -2.7970068 ,
              2.0984807 ,  6.2855573 ,  2.6990292 ,  0.7760899 , -1.0006282 ,
             -1.6409808 , -3.4258785 , -0.11701375,  3.620262  , -4.079018  ,
              4.0971594 , -3.2509642 ,  0.3097818 ,  3.5571876 , -0.7968726 ,
             -2.8652878 ,  5.1031704 ,  1.6793871 , -2.1226752 ,  5.986605  ,
              0.15131441, -0.08931142, -4.5201797 , -4.2452917 ,  8.549718  ,
              4.471525  ,  0.99099433, -2.8955932 , -1.3617678 , -3.047677  ,
              0.16512078,  3.045219  ,  0.28825712,  1.2651473 , -2.7130852 ,
             -0.7071771 ,  0.73940206, -7.5693707 , -5.2546077 , -2.6441014 ,
              3.062767  , -1.0616007 ,  3.0221672 , -4.145956  , -6.2783084 ,
              0.28456658, -0.04906318, -2.1165097 , -1.3185437 ,  4.7343583 ,
              0.7987691 , -0.89384216, -1.7637501 ,  1.5510547 , -5.4871173 ,
              6.088364  , -0.5114596 , -1.7852409 ,  2.9882193 , -2.160759  ,
              0.2773401 ,  3.7901676 , -2.3185658 , -0.627069  ,  2.323418  ,
             -2.0181699 ], dtype=float32)
```

WORD EMBEDDING

Généralement de telles représentations sont construites via l'entraînement d'un réseau de neurones sur des tâches de prédiction.

WORD EMBEDDING

Concrètement un apprentissage est effectué, sur des corpus **gigantesques** de textes via ces réseaux,

- soit pour prédire un mot en fonction du contexte,
- soit pour prédire le contexte en fonction du mot.

N-GRAMS

De nombreux mots, en anglais comme en français, disposent d'un préfixe ou d'un suffixe qui viennent en changer le sens. En particulier un préfixe peut inverser le sens d'un mot. (correct et incorrect)

À l'inverse, un suffixe différent n'affecte généralement pas le sens global du mot. (jour, journée)

N-GRAMS

Il serait pertinent de considérer un *embedding* qui prendrait en compte ces caractéristiques.

L'idée est de mettre le plus loin possible, dans l'espace vectoriel considéré, les mots opposés et de mettre à proximité les mots de sens similaires. Une façon de faire cela, tout en évitant de gérer explicitement, des règles linguistiques complexes est de considérer des n-grams.

N-GRAMS

Un n-gram est une sous-séquence de *n* éléments (ici des lettres) dans une séquence donnée.

Par exemple, dans la langue française, le 2-gram (ou bi-gram) le plus répandu est « de », qui apparaît à la fois dans l'article « de » mais aussi dans des mots comme « monde » ou « mode ».

N-GRAMS

Un n-gram est une sous-séquence de *n* éléments (ici des lettres) dans une séquence donnée.

Par exemple, dans la langue française, le 2-gram (ou bi-gram) le plus répandu est « de », qui apparaît à la fois dans l'article « de » mais aussi dans des mots comme « monde » ou « mode ».

N-GRAMS

Un modèle possible est d'effectuer un apprentissage, possiblement en brute force (*ie* en testant toutes les possibilités permises), sur l'ensemble de ces n-grams pour en calculer une représentation vectorielle pertinente. C'est le type de modélisation proposée par *fastText*

N-GRAMS

Un modèle possible est d'effectuer un apprentissage, en brute force (*ie* en testant toutes les possibilités permises), sur l'ensemble de ces n-grams pour en calculer une représentation vectorielle pertinente. C'est le type de modélisation proposée par *fastText*

N-GRAMS

L'idée est de considérer chaque mot comme l'ensemble des n-grams qui le constitue, plus le mot lui-même.

Par exemple, pour *n*=3, le mot « *where* » est constitué des éléments suivants « _wh », « whe », « her », « ere », « re_ » et « where ».

L'*embedding* du mot correspond alors à la somme des tous les vecteurs associés à l'ensemble des n-grams qui le constituent.

EMBEDDING DE PHRASES

Il est parfois plus utile de manipuler plus que des mots notamment lors d'une classification de texte ou d'une génération automatique de résumés.

EMBEDDING DE PHRASES

Il existe alors plusieurs façons de procéder :

- faire la moyenne de l'embedding associé aux mots de la phrase ;
- effectuer une moyenne pondérée (par exemple avec les poids calculés via procédure TF-IDF, une mesure statistique caractérisant l'importance, en nombre d'occurrences, du mot dans un texte ou un corpus donné).

EMBEDDING DE PHRASES

[…] De telles approches sont possibles mais ne sont pas toujours les mieux adaptées. En effet, elles reviennent à utiliser *a posteriori* un *embedding* de mots qui a été entraîné dans un contexte différent de celui pour lequel l'*embedding* de phrase est recherché.

EMBEDDING DE PHRASES

Une alternative est d'effectuer la constitution de l'*embedding* de phrases lors de la phase d'entraînement, voir par exemple Sent2Vec. Dans ce cas l'objectif est de représenter les phrases sous la forme d'une somme de sous-phrases, de façon similaire aux n-grams de caractères mais avec des mots.

EMBEDDING DE PHRASES

Il existe également une extension du modèle de *Word2vec*, disponible et nommé *Doc2Vec*, qui permet également d'entraîner un *embedding* de textes directement sur un corpus de textes donnés.