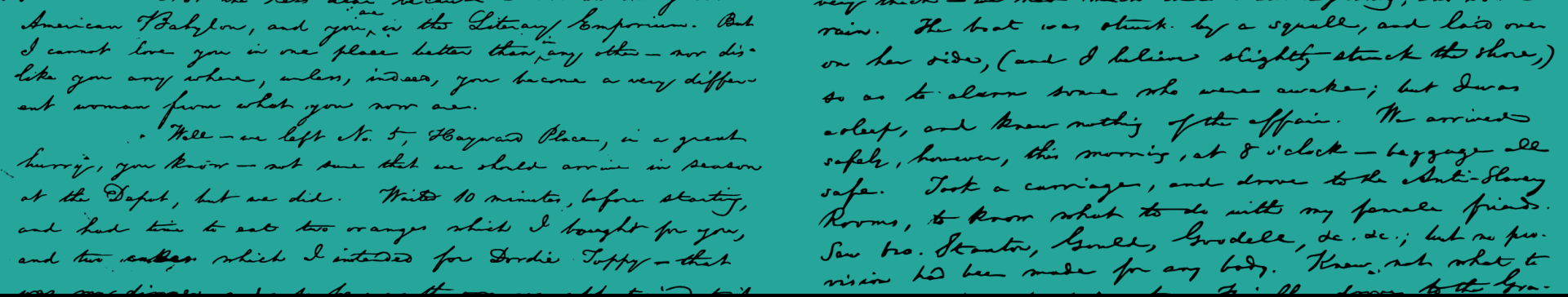


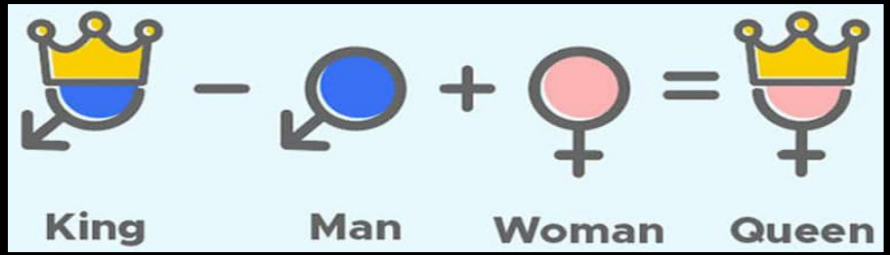
Les méthodes du **word embedding**

—

Par un SID pour un SID

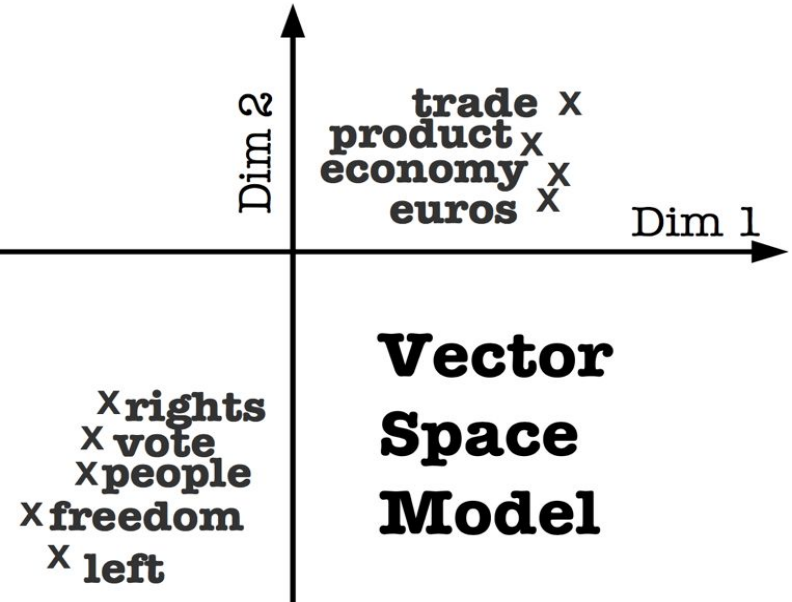
word embedding

ensemble de techniques de machine learning qui visent à représenter des **données textuelles** (mots ou phrases d'un texte) par des **vecteurs de nombres réels**, décrits dans un modèle vectoriel (Vector Space Model).



Quelles sont ces techniques ?

Les méthodes classiques



- TF-IDF (pondération des termes, stopwords)
- Word2Vect (représentation mot-contexte,s)
- GloVe (Global Vector for Word Representation)

MÉTHODE DU TF-IDF

Utilisé dans une stratégie de référencement afin de déterminer les mots-clé et les termes qui augmentent la pertinence des textes analysés.

1. **TF** - Term Frequency

= Fréquence d'apparition d'un terme dans un doc p/r aux autres termes contenus dans le doc. Présence du logarithme pour une meilleure distribution des poids et ainsi augmenter la signification de la valeur mesurée.

1. **IDF** - Inverse Document Frequency

= Mesure de signification d'un terme en fonction de sa distribution et de son utilisation dans le corpus de doc.

1. **TF-IDF** - Fréquence Relative des termes d'un doc p/r à tous les autres docs du corpus

= Fréquence réelle des termes et son potentiel pour optimiser le texte existant.

Utilisation de scikit-learn

FORMULAIRE DE RAPPEL DES FORMULES

+ TF - Term Frequency

$$TF(i) = \frac{log_2(Freq(i,j) + 1)}{log_2(L)}$$

Freq(i,j) = Fréquence du mot i dans le doc j - L = nombre total de mots dans le doc j

+ IDF - Inverse Document Frequency

$$IDF(i) = \log\left(\frac{N_D}{f_i} + 1\right)$$

N_D = Nombre total de doc dans le corpus - f_i - Nombre de tous les docs dans lequel le mot i apparaît

+ TF-IDF - Fréquence Relative des termes d'un doc p/r à tous les autres docs du corpus

$$TF(i,j) = TF_{i,j} * IDF_i$$

Avantages de l'analyse TF-IDF	Inconvénients de l'analyse TF-IDF
donne une grande chance de découvrir le bourrage de mots-clés existant	examine toujours le contenu rédactionnel complet d'un document
privilégie la pertinence et la singularité en tant que critères décisifs pour la pondération des fréquences	ne fournit pas d'informations sur les paragraphes ou passages précis qui ont besoin d'être optimisés
évalue mieux les mots-clés avec une concurrence moindre que ceux avec une forte concurrence	ne convient pas aux textes courts contenant peu de mots
combine les disciplines de l'analyse spécifique aux documents et de l'analyse générale	difficile à utiliser dans les processus de travail où la rapidité et la réactivité sont requises
aplanit les résultats en utilisant des logarithmes pour obtenir des données plus pertinentes	difficile de déterminer avec précision le nombre de tous les documents pertinents

MÉTHODE Word2Vect

Grandement utilisée, cette méthode propose deux architectures neuronales. L'entraînement du réseau de neurone se fait en parcourant tout le texte et en modifiant les poids des mots pour réduire l'erreur de prédiction de l'algorithme.

1. **CBOW** - Continuous Bag Of Words model

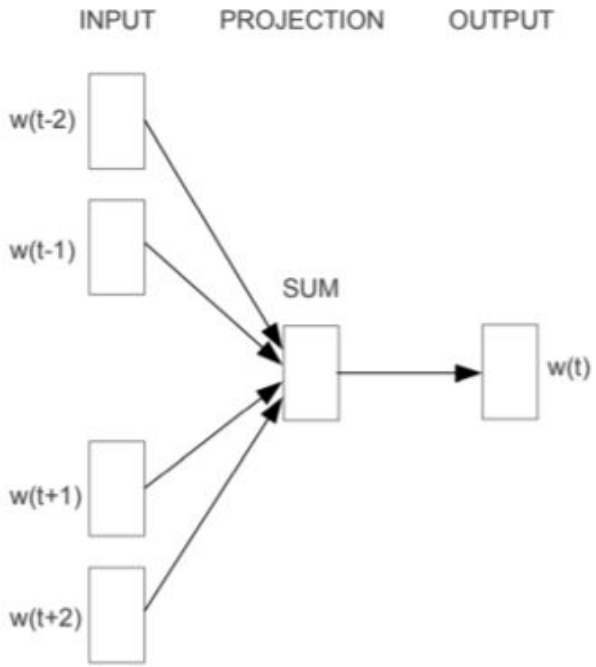
= reçoit en entrée le contexte d'un mot (les termes qui l'entourent dans une phrase) pour prédire le mot.

1. **SKIP-GRAM**- Inverse Document Frequency

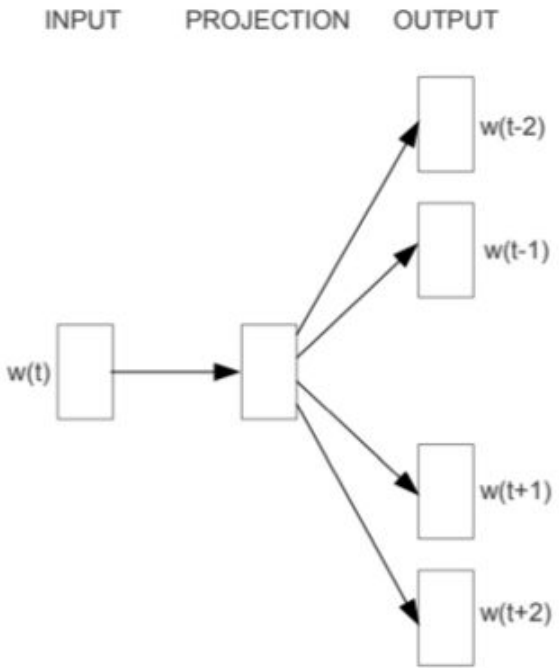
= reçoit en entrée un mot et essaye de prédire son contexte.

Paramètres à prendre en compte : la dimensionnalité de l'espace vectoriel (= nombre de descripteurs numériques utilisés pour décrire le mot ~100-1000), la taille du contexte du mot (de taille n-gram).

Librairies python référence : gensim, spacy

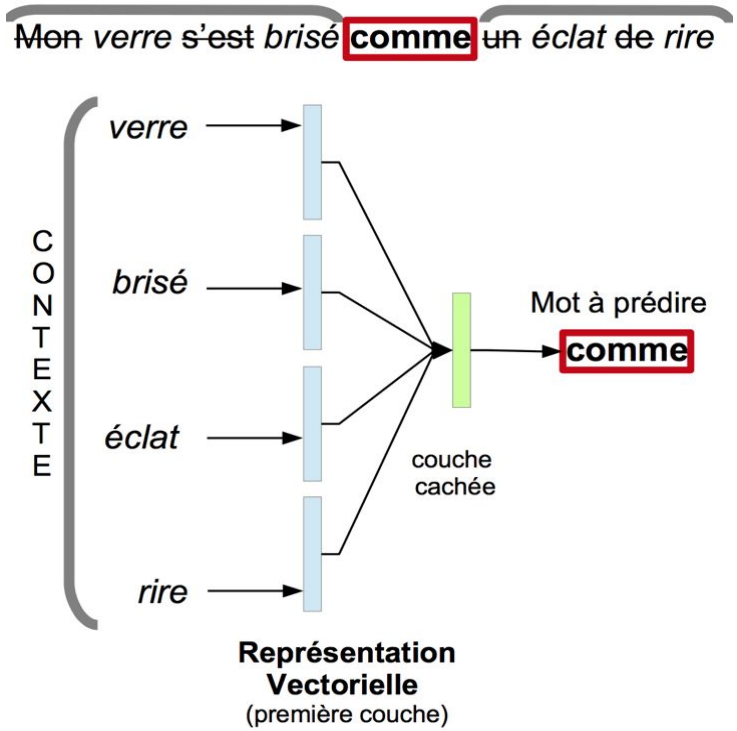


CBOW

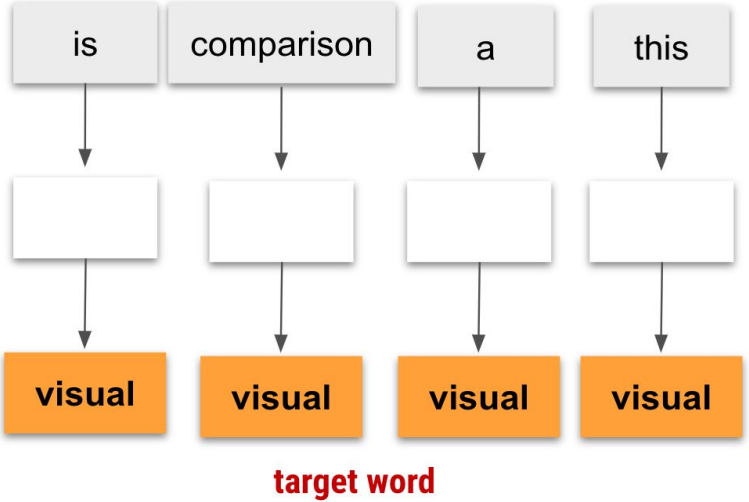


Skip-gram

Architecture CBOW



SkipGram



This is a visual comparison

MÉTHODE GloVe

Modèle d'apprentissage non supervisé qui prend en compte toute l'information portée par le corpus et non pas la seule information portée par une fenêtre de mots, d'où le nom GloVe, pour VEcteur GLObal.

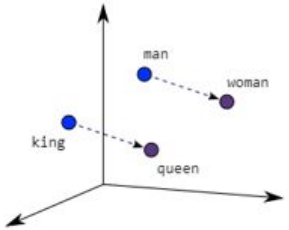
1. **Matrice de cooccurrence des mots** - Matrice MG

= Chaque élément MGij représente le nombre de fois où le mot mj apparaît dans le contexte du mot mi. Contexte = fenêtre glissante.

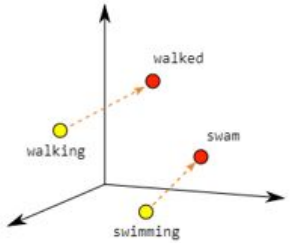
1. **Modèle de régression par moindres carrés -** Construction des représentations vectorielles globales pour chaque mot

Librairie python référence : gensim

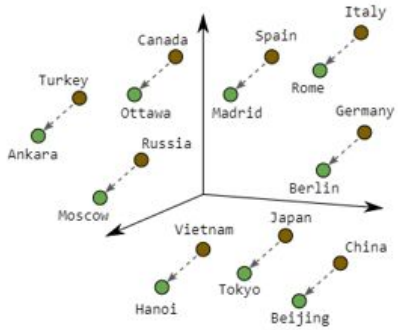
Word2Vec



Male-Female

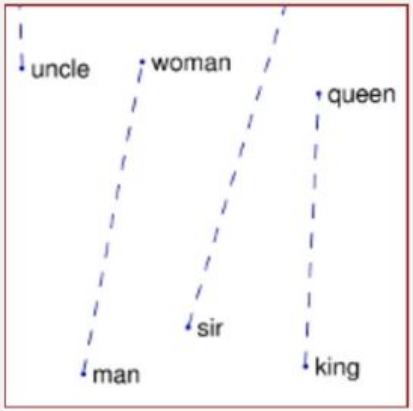


Verb Tense

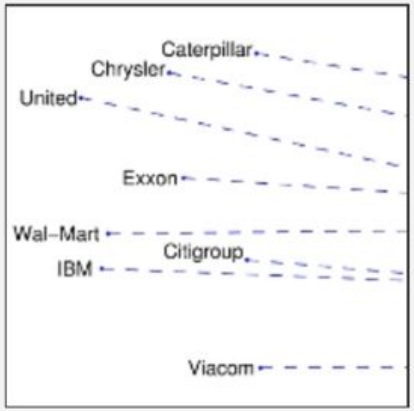


Country-Capital

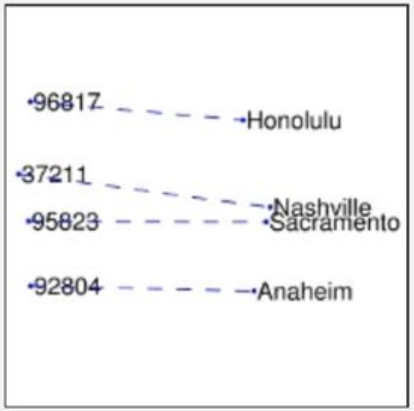
GloVe



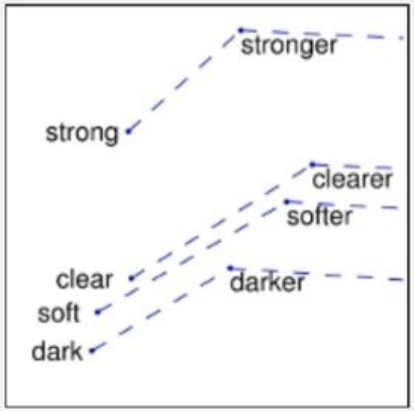
man - woman



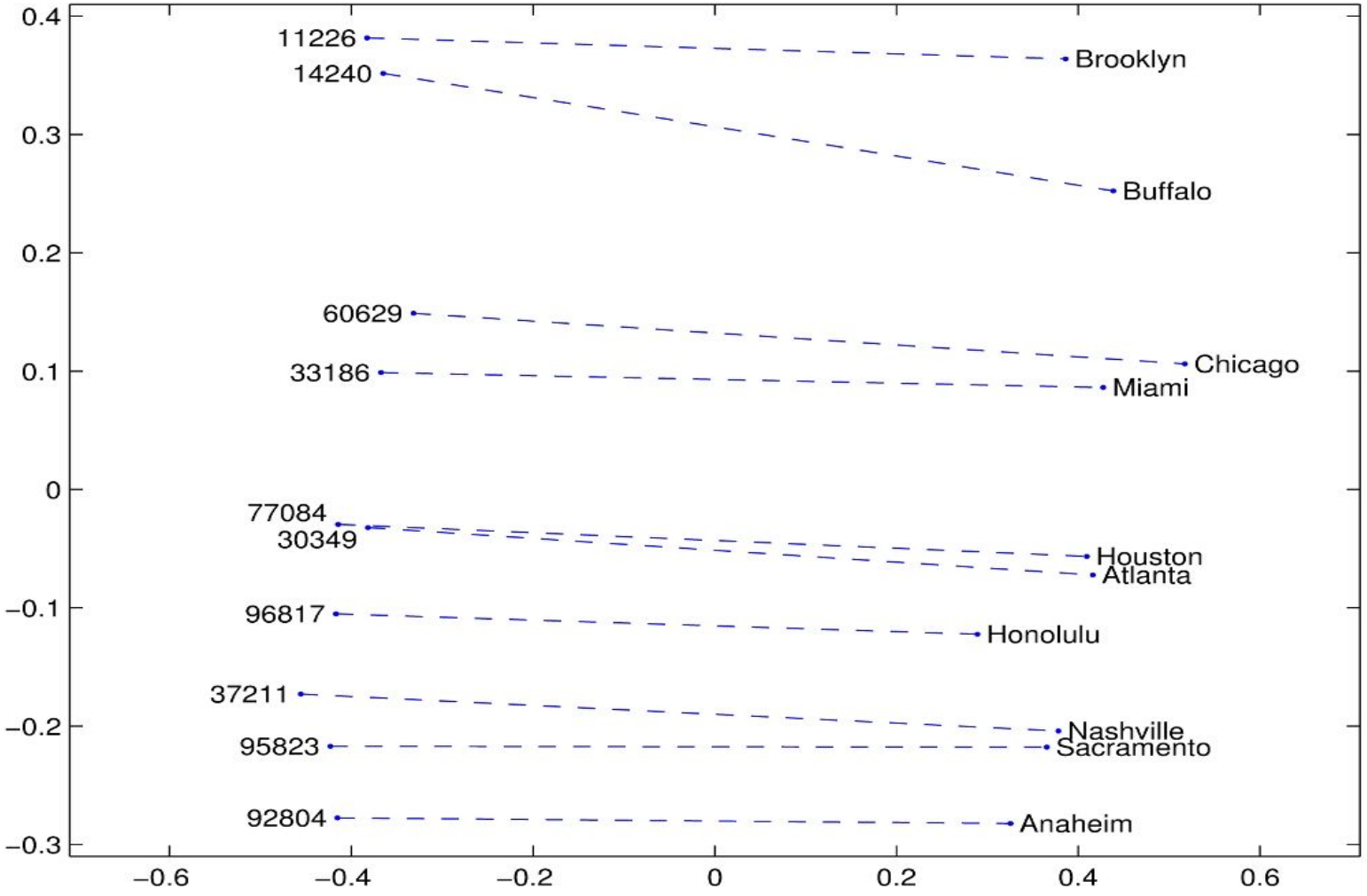
company - ceo



city - zip code



comparative - superlative


11226
Brooklyn
14240
Buffalo
60629
Chicago
33186
Miami
77084
30349
Houston
Atlanta
96817
Honolulu
37211
Nashville
95823
Sacramento
92804
Anaheim
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.6
−0.4
−0.2
0
0.2
0.4
0.6

Similarity with GloVe

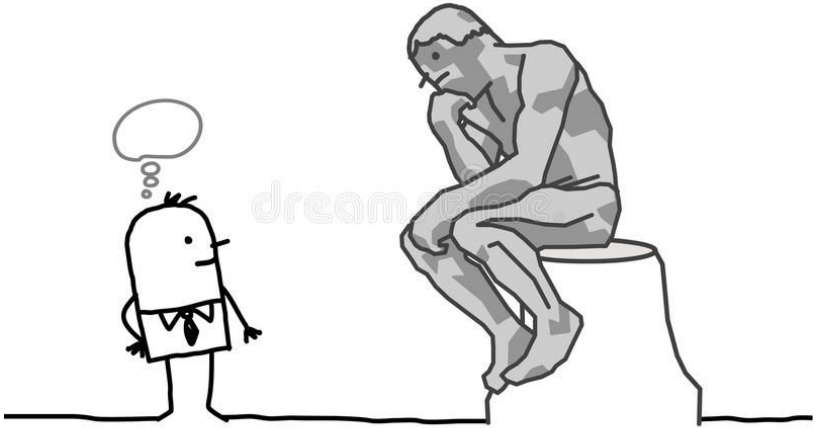
	phrase	score
0	barrack obama	0.956801
1	barrack h. obama	0.944671
2	barrack hussein obama	0.937000
3	michelle obama	0.905201
4	donald trump	0.729601
5	melania trump	0.614963

Similarity with Jaccard

	phrase	score
0	barrack obama	0.249377
3	michelle obama	0.249377
1	barrack h. obama	0.199601
2	barrack hussein obama	0.199601
4	donald trump	0.000000
5	melania trump	0.000000

Phrases most similar to “Barack Hussain Obama”

Autres méthodes



- FastText (Enrichir les vecteurs de mots avec des informations sur le mot ; extension du modèle Word2Vect)
- Poincaré Embeddings (utilisation de la géométrie hyperbolique pour capturer les propriétés hiérarchiques du mot que l'espace euclidien ne permet pas)

Quelle est la finalité ?

Amélioration des performances des méthodes de traitement automatique des langues (NLP - Natural Language Processing)

ex : Sentiment analysis, topic modeling

Liens pratiques

¤ Idées générales du WE.

¤ WE méthodes + code python

¤ Overall des méthodes

¤ Méthode GloVe highlight

¤ Évaluation des représentations vectorielles

¤ WE neural networks