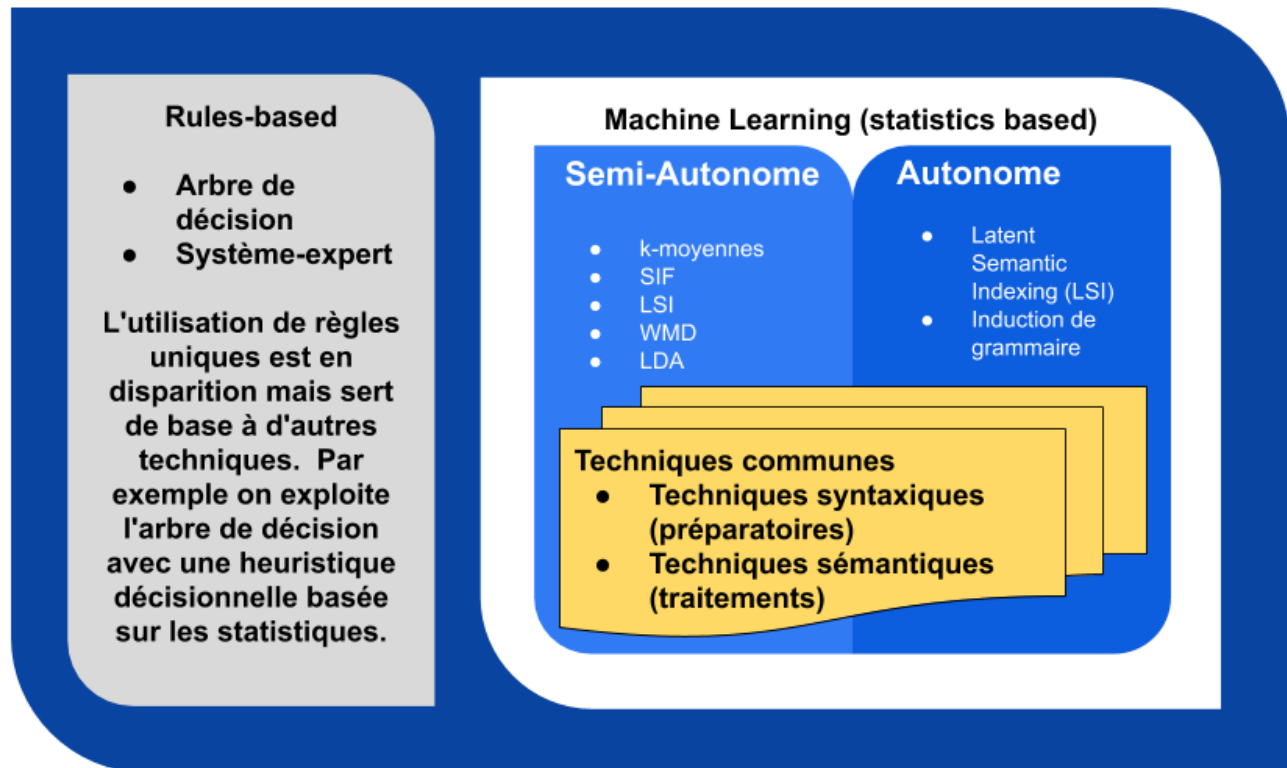


ALGORITHMES DU NLP

CLASSIFICATION DES ALGORITHMES

Bien que d'une part l'origine du NLP ait utilisé les deux sortes d'algorithmes, de nos jours c'est plutôt le NLP qui est devenu une des catégories de l'intelligence artificielle (Machine learning). De nos jours on dit que le "[Natural Language Processing or NLP is a field of Artificial Intelligence that gives the machines the ability to read, understand and derive meaning from human languages.](#)"



FONCTIONNEMENT GÉNÉRAL D'UNE ANALYSE NLP

Analyse syntaxique :

- Découper le texte en mots
- Représenter des suites de mots en vecteurs binaires. Certains vecteurs 'flag' la présence de certains mots ou non. D'autres vecteurs indiquent la position de chaque mot dans le texte.

					juste concernant enfant, étiquetage de la position			
roi	reine	homme	femme	enfant	position 0	position 1	position 2	position 3
0	0	0	0	1	0	1	0	0

Source : <https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/>

Analyse sémantique :

Source : <https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32>

- Named entity recognition (NER): It involves determining the parts of a text that can be identified and categorized into preset groups. Examples of such groups include names of people and names of places.
- Word sense disambiguation: It involves giving meaning to a word based on the context.
- Natural language generation: It involves using databases to derive semantic intentions and convert them into human language.

ALGORITHMES

RÉSUMÉ DES ALGORITHMES D'ANALYSE SYNTAXIQUE

Sources : https://en.wikipedia.org/wiki/Natural_language_processing,
<https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32>

- Découpage (tokenization) = découper le texte en mots, retirer les mots d'arrêt, la ponctuation
- Parsing = analyse de la syntaxe avec un Parse-Tree
- Lemmatisation & racinisation (lemmatization, stemming) = réduire les mots à leur lemme de base afin de les rendre comparable, par exemple ami et amitié deviennent tous les deux ami (Source : <https://medium.com/@adriensieg/text-similarities-da019229c894>)
- Induction de grammaire = déduction automatisée des règles de grammaire selon les observations d'un texte (sans règles préalables) - pour les algorithmes autonomes seulement
- Segmentation en Morphemes = Approche inutile en anglais et impossible en indien, qui consiste à découper les mots selon leurs composants étymologiques.
- Étiquetage contextuel (Part-of-speech tagging) = Classification des mots en tenant compte de leur contexte. Le même mot peut être un adjectif ou un nom dans des parties de phrases différentes, des mots identiques peuvent avoir des significations opposées selon le contexte par exemple 'chat' qui est un animal ou une discussion en ligne. Un mot pourrait être mal orthographié mais être sonorement identique à un autre et le contexte permet de le déterminer, par exemple dans **Je suis mêler**, le contexte montre que ce n'est pas l'infinitif mais le participe passé malgré la faute.

RÉSUMÉ DES ALGORITHMES D'ANALYSE SÉMANTIQUE

Source : <https://medium.com/@adriensieg/text-similarities-da019229c894>

- Similarité de Jaccard = méthode très limitée qui consiste à calculer le ratio de mots sémantiquement identiques ou similaires par rapport au nombre de mots au total, cette méthode ne tient pas compte de l'ordre des mots par exemple **L'ours mange le singe** et **Le singe mange l'ours** seraient sémantiquement identiques selon Jaccard !
- Partitionnement en k-moyennes (k-mean clustering) (<https://fr.wikipedia.org/wiki/K-moyennes>)
 - Sac de mots : Count Vectorizer method
 - Insertion de mots : [Word2Vec](#), [Doc2Vec](#), [Glove](#), etc. - <https://yidatagithub.io/2017-08-03/word2vec-aha/>
 - CBOW = Continuous Bag of words = pour détecter un mot manquant
 - [Skip-Gram](#) = pour prédire les mots alentour d'un mot donné, donc prédire la suite d'une phrase
- Source : <https://yidatagithub.io/2017-08-03/word2vec-aha/>
- SIF (Smooth Inverse Frequency) - certaines phrases avec des mots similaires ont des sens tout à fait différents si ce n'est inverse. SIF essaie de corriger ce problème en attribuant des poids aux mots. Cela retire la variation de syntaxe non-pertinente en sémantique.
- BERT (Bidirectional Encoder Representations from Transformers) - il permet de mieux tenir compte de tout le contexte d'un mot dans une phrase parce qu'il est en fait non-directionnel au lieu de parser séquentiellement. <https://arxiv.org/pdf/1706.03762.pdf>
<https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>
- Latent Semantic Indexing (LSI) = Algorithme qui réduit les dimensions avant la classification finale par une projection intelligente dans un espace de dimension plus limité - On résout une dimension à la fois - [Comme un thesaurus auto-généré](#).
- Similarité de cosinus = Méthode de calcul de similarité sur des vecteurs de mots - Une telle méthode s'applique donc aux groupes de mots après classification des mots individuels dans un espace sémantique similaire
- WMD (Word Mover Distance, [transport optimal](#)) = minimisation de la distance sémantique à parcourir par l'estimé de sémantique du vecteur de mot pour rejoindre l'autre estimé de sémantique de l'autre vecteur
- [LDA with Jensen-Shannon distance](#) = mesure la similarité entre deux distributions de probabilités (donc deux groupes de données) ce qui est utile pour comparer des documents dans leur intégralité ou pour détecter des sujets peu évidents dans les documents.
- Encodeurs de phrases = Encode les phrases dans des vecteurs spécifiques en vue d'analyses spécifiques. (Facebook + Google)
 - Auto-Encodeur = Permet de reconstruire l'original après l'avoir décomposé pour analyse, confirmant si l'analyse était correcte. (Source Encodeurs <https://arxiv.org/pdf/1706.03762.pdf>)