République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université Larbi Tébessi - Tébessa
Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie
Département : Mathématiques et Informatique

Mémoire de fin d'étude
Pour l'obtention du diplôme de MASTER
Domaine : Mathématiques et Informatique
Filière : Informatique
Option : Systèmes d'information

Thème

Arabic Word Embedding model hotels reviews

Présenté Par :
GASMI FAOUZI

Devant le jury :

Mr Laouar M.R	*Prof*	*Univérsité Larbi Tébessi*	*Président*
Mr Sahraoui A	*MCB*	*Univérsité Larbi Tébessi*	*Examinateur*
Mr Bendib I	*MCB*	*Univérsité Larbi Tébessi*	*Encadreur*

Date de soutenance : Juillet 2019

Résumé

L'arrivée des réseaux sociaux a conduit à des développements dans des domaines tels que l'analyse des sentiments, la vision par ordinateur, la reconnaissance de la parole et le traitement automatique des langages naturels (TALN). L'un des développements récents les plus influents du T.A.L.N est l'utilisation de mots incorporés autrement dit le Word embedding, où les mots sont représentés comme des vecteurs dans un espace continu, capturant de nombreuses relations syntaxiques et sémantiques entre eux. Dans ce mémoire nous allons réaliser un model Word embedding pour la langue arabe spécifique pour le lexique du domaine de l'hôtellerie.

La langue arabe présente un manque flagrant dans l'utilisation de ces technologies soit du côté de ressources ou travaux réalisés.

Notre représentation distribuée de mots (model Word embedding) vise à fournir à la communauté de recherche arabe en TALN notamment la tache de l'analyse des sentiments

Les mots clés : Réseaux sociaux, Analyse des sentiments, Traitement Automatique des Langages Naturels, Word embedding, langue arabe, hôtellerie.

Abstract

The arrival of social networks has led to developments in areas such as sentiment analysis, computer vision, speech recognition and natural language processing (NLP). One of the most influential recent developments in NLP is the technic of Word embedding, where words are represented as vectors in a continuous space, capturing many syntactic and semantic relationships between them. In this thesis we are going to build a Word embedding model for the Arabic language specific to the lexicon of the field of the hotel industry.

The Arabic language shows a flagrant lack of use of these technologies either in terms of resources or work done.

Our distributed representation of words (model Word embedding) aims to provide to the Arab research community in TALN including the task of the analysis of feelings

Keywords : Social networks, Sentiment analysis, Natural language Processing, Word embedding, The Arabic Language, Hotel.

ملخص

أدى التقدم في استعمال شبكات التواصل الاجتماعي إلى احداث تطورات في مجالات مثل تحليل المشاعر ، رؤية الكمبيوتر ، التعرف على الكلام ومعالجة اللغات الطبيعية . أحد أهم التطورات الحديثة في معالجة اللغات الطبيعية هو استخدام الكلمات المضمنة (تقنية تضمين الكلمات)، حيث يتم تمثيل الكلمات كأشعة في مساحة مستمرة ، هذه التقنية (تضمين الكلمات) تسمح بالتقاط العديد من العلاقات النحوية والدلالية بينها. في هذه الأطروحة ، سنعمل على بناء نموذج لتضمين الكلمات للغة العربية خاص بمعجم مجال الفندقة.

تُظهر اللغة العربية نقصًا صارخًا في استخدام هذه التقنيات سواء من حيث الموارد أو الاعمال المنجزة في هذا الاطار.

يهدف تمثيلنا الموزع للكلمات (نموذج تضمين الكلمات) إلى توفير نموذج لتضمين كلمات اللغة العربية هذا الاخير يمكن استخدامه في مختلف ميادين معالجة اللغات الطبيعية ،خاصةمنها تحليل المشاعر.

الكلمات المفتاحية: شبكات التواصل الاجتماعي، تحليل المشاعر ، معالجة اللغات الطبيعية ،تضمين الكلمات، اللغة العربية، الفندقة

.

Remerciements

Je remercie mes très chers parents, qui ont toujours été là pour moi, mes chers frères et sœurs pour leur encouragement.
J'adresse mes remerciements à tous mes enseignants, notamment mon encadrant M. BENDHIB ISSAM pour sa disponibilité, sa patience, sa compréhension, ses qualités humaines et ses intérêts portés pour notre sujet de travail. Je le remercie de m'avoir fait confiance et d'avoir été présent aussi souvent que possible malgré ses tâches pédagogiques.
Mes remerciements vont aussi à
Prof Laouar M.R , d'avoir ménagé son temps pour présider ce jury
Dr. Sahraoui.pour avoir bien voulu siéger dans ce jury afin d'examiner et critiquer ce mémoire et nous éclairer par ces précieux conseils.
Dr. Bendib. I aucun remerciement ne saurait exprimer notre respect et considération pour les orientations que vous avez consenties pour notre étude de l'Université.
Je tiens à exprimer toute ma reconnaissance à Monsieur, le directeur de la C.N.R-Tébessa, AOULMI FATEHI, qui m'a permis de continuer mes etudes universitaires
Je remercie, mes amis, notamment ceux qui m'ont aidé dans la réalisation de ce travail.
Enfin, Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui m'ont aidé lors de la realisation de ce projet.

Table des matières

Liste des figures

Liste des tableaux

Introduction Générale

Les médias sociaux, blogs, forums, sites Web de commerce/service électronique, etc. encouragent les citoyens à partager leurs opinions, leurs émotions et leurs sentiments publiquement.

Les opinions des gens sont des informations très précieuses pour la prise de décision, et en tirer des avantages d'expérience, le contenu accumulé des opinions doit être extrait et analysé correctement. Les informations tirées sont très utiles pour le consommateur ainsi que pour les fabricants, également pour les dirigeants ou les décideurs du pays.

Le problème considéré par tous les *consommateurs d'opinions*[1] est qu'il y a une telle richesse des textes à traiter, et qu'il est difficile de tout lire, ce qui pose le problème d'exploiter ces ressources en temps et en coût très élevé pour avoir ce qui est nécessaire, d'où la nécessité de traitement des textes bruts et l'extraction des expressions pertinentes qui peuvent être subjectives ou objectives.

Les techniques développées pour exploiter ces ressources afin d'aider les organismes et les individus à obtenir les informations importantes facilement et rapidement. Ces techniques sont les fruits des travaux de recherche dans plusieurs domaines notamment l'analyse des sentiments.

Pour exploiter ces ressources d'une façon optimal, il faut les représenter correctement, beaucoup de techniques de représentation ont été développés entre autre la représentation vectorielle. Cette représentation permet d'un traitement rapide et efficace des grands corpus de données.

L'hôtellerie, appelée également industrie hôtelière, est une activité qui regroupe l'ensemble des établissements qui proposent un service d'accueil — de gîte et/ou de couvert — à des clients, de passage ou locaux, pendant une durée déterminée, en échange d'une contribution.

La Planification de voyage et réservation d'hôtel sur site est devenu l'un des plus importants un usage commercial. Les dernières années ont vu une croissance rapide du commerce en ligne groupes de discussion et sites de révision (par exemple, www.tripadvisor.com), comme les autres services et produits proposés, celui de l'hôtellerie peut tirer plein d'avantage de l'analyse des sentiments.

[1] Ceux qui utilisent les opinions d'autres individus ou d'autres organismes, par exemple pour améliorer la qualité d'un service ou d'un produit

Problématique

La recherche d'informations utiles sur les avis d'utilisateurs avant de réserver ou de choisir un hôtel plutôt qu'un autre est devenue une pratique courante pour de nombreuses personnes.

Pour une meilleure analyse des avis des internautes, par l'analyse des sentiments, il faut mener une stratégie solide pour la représentation de ces données /information tout en gardant leur valeur sémantique.

Le contenu du web à exploiter est très volumineux, varié, et pose des problèmes au niveaux linguistique et sémantique, d'où il est nécessaire de le traiter et d'en tirer un vocabulaire solide ce qui va faciliter la tâche d'analyse, et donne des résultats optimaux.

En outre des problèmes mentionnés, la langue arabe présente ses propres difficultés, ces difficultés sont liés à l'arabe autant que lange (structure, morphologie, dialecte …..)

Pour la langue arabe ce domaine de représentation /analyse est encore presque vierge, et ça se traduit par le nombre très modeste des Data set et travaux réalisés dans ce domaine

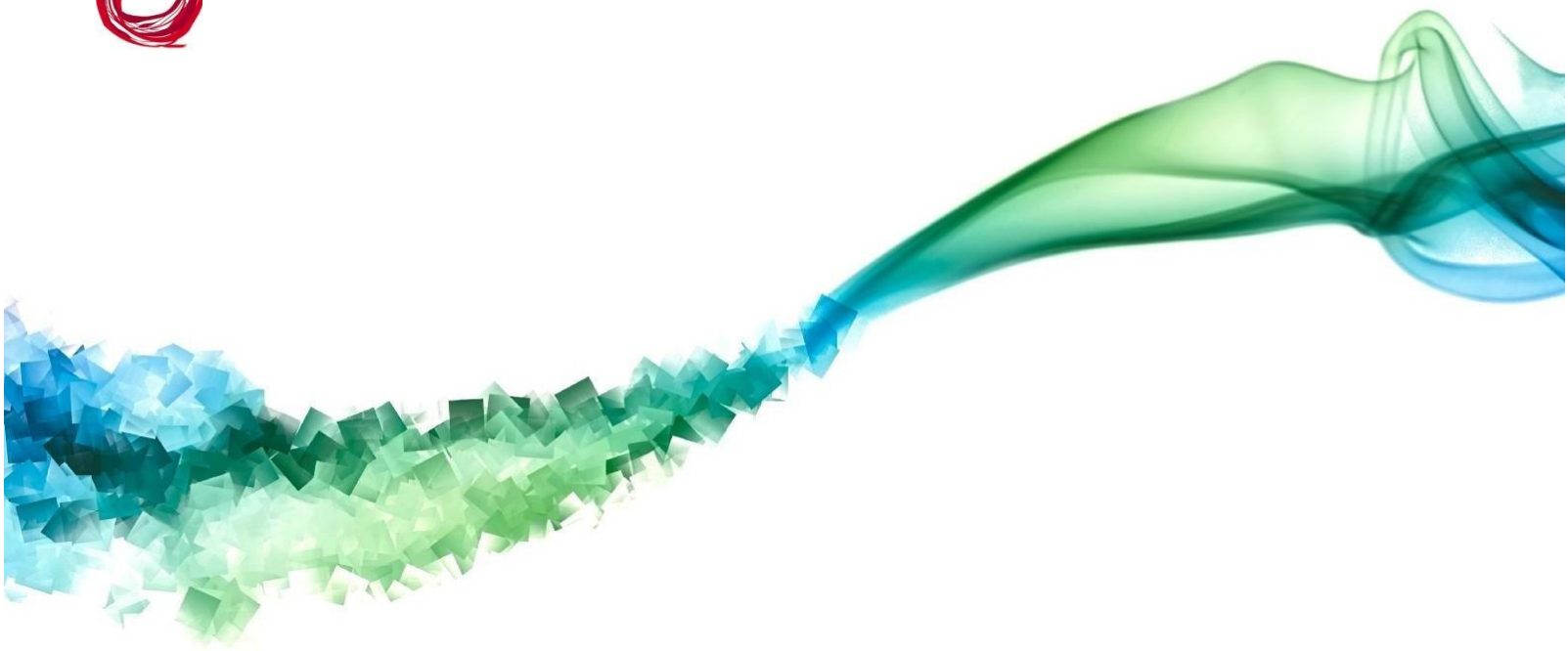
Objectifs

- La construction d'un corpus des avis rédigés en arabe relatifs au domaine de l'hôtellerie.
- La construction d'une base de vocabulaire arabe solide relatif à l'hôtellerie qui peut être utilisée dans les différents tâches de l'Analyse des sentiments, Traitement Automatique des Langages Naturels……etc.
- La construction d'un model Word Embedding efficace pour la représentation du vocabulaire construit.

Organisation du manuscrit

Ce document commence par une introduction générale sur notre travaille, il contient trois chapitres : le premier chapitre aborde le concept de l'analyse des sentiments, le deuxième sur les techniques Word Embedding / Word2Vec et les travaux connexes pour la langue arabe, le troisième contient notre contribution (stratégie, construction du model et évaluation) et finalement une conclusion générale.

Chapitre 01



Analyse des Sentiments

1. Introduction

Le web2.0 appelé également le web participatif ou encore interactif a permis à l'internaute de dépasser le rôle du simple spectateur et devenir un acteur dans le contrôle de l'information (collaborer, partager, réagir, donner son opinion …), par conséquence on trouve sur le web une énorme quantité de données textuelles produite par les visiteurs des réseaux sociaux, des blogs, des forums, sites de commerce électronique, etc. Ces textes reflètent le plus souvent les opinions des internautes vis-à-vis des produits, des services, des films, des musiques, des articles, des lieux etc.

Les commentaires publiés librement par les internautes sont stratégiques pour les entreprises, les chercheurs, les économistes, les politiciens, …etc, ils représentent une véritable mine d'or d'informations, toutefois, la nature de ces messages du langage naturel, les différencie des données que les entreprises et les communautés avaient l'habitude de traiter et apporte de nouvelles contraintes d'analyse ce qui rend les méthodes classiques de la recherche d'information peu efficaces.

L'extraction, l'analyse et l'exploitation de de ces massives quantités de données (le plus souvent non structurés) vont au-delà du pouvoir humain et du temps raisonnable, d'où la nécessité de nouveaux outils de traitement automatique du langage naturel.

De nombreux systèmes d'analyse ont été développés pour permettre d'optimiser l'exploitation des dédites données afin d'offrir aux entreprise une connaissance clients d'avantage poussée, et de garantir au chercheurs un système de prédiction et d'étude rigoureuse. Parmi ces systèmes, ceux spécialisés dans le traitement automatique du langage naturel, et en particulier l'analyse de sentiments, ils sont développés afin d'identifier rapidement les sentiments clés issus d'une colossale quantité de données textuelles échangées sur la toile.

2. Bref historique

Peu après l'apparition du web 2.0 avec son caractère dynamique qui a abouti à l'apparition du data relatif aux interactions des internautes, au début des années 2000 la nécessité d'exploiter ces données a conduit à l'apparition du domaine du l'analyse des sentiments, le terme lui-même a apparu pour la première fois dans l'article de Dave et al **[1]**. L'année 2005 semble marquer le début d'une prise de conscience généralisée des problèmes de recherche et opportunités que l'analyse des sentiments peut offrir.

Les facteurs à l'origine de cette " explosion" comprennent :

-La montée en puissance des méthodes d'apprentissage automatique du traitement du

langage naturel et de la recherche d'informations.

-La disponibilité d'ensembles de données pour l'apprentissage des algorithmes d'apprentissage automatique.

-La réalisation des défis intellectuels fascinants et des applications commerciales.

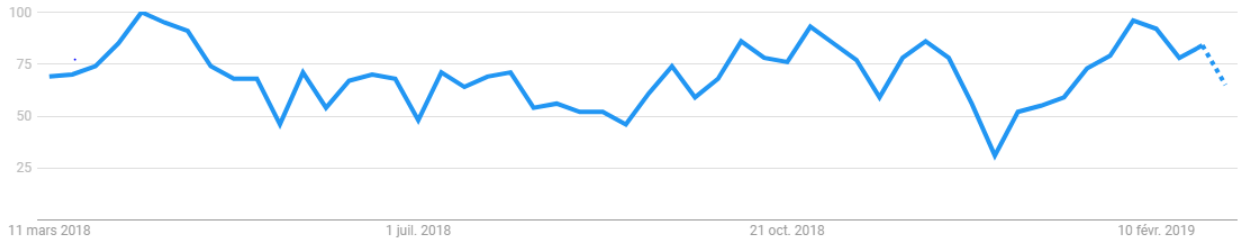


Fig.1.1 Tendance de l'analyse des sentiments au cours des 10 dernières années[1]

3. Définitions

3.1. Opinion

D'après le dictionnaire LAROUSSE « opinion : emprunté du latin opinio {opinion, conjecture, croyance} : est un jugement, qu'un individu ou un groupe émet sur un sujet, des faits, ce qu'il en pense. Synonymes : avis, conviction, sentiment, idée, impression, pensée, point de vue ».

3.2. Faits & Opinions

Le texte autant qu'une suite des informations comporte deux catégories principales pour classer ces informations textuelles : faits et opinions. Dans le premier cas, il s'agit de descriptions objectives (énoncé) sur les entités et les événements dans le monde, dans l'autre cas, il s'agit d'expressions subjectives (opinion, sentiment, évaluation ou jugement ……) d'un individu à propos d'un objet ou d'un sujet particulier **[2]**.

Le plus souvent ils existent des indices textuels qui permet la distinction objectivité/subjectivité du texte.

[1] Source de photo: Google Trends3 (www.google.com/trends)

Objectif	Subjectif
Un style, une forme et un vocabulaire neutre.	Un style, un ton et un vocabulaire descriptifs, expressifs, appréciatifs.
L'emploi de pronoms personnels à la troisième personne, comme «il » ou « on », sauf à l'intérieur des citations où ils ne sont pas obligatoires.	L'emploi de pronoms personnels de la première et la deuxième personne à l'intérieur comme à l'extérieur des citations : «je », « tu », « nous » et « vous ».
L'utilisation de citations, des affirmations, des références et des statistiques renforcées.	L'utilisation de citations pour renforcer des opinions ou des jugements.
La construction des phrases déclaratives. Les phrases interrogatives directes, exclamatives et impératives sont rares.	L'emploi de la phrase exclamative suivie de sa justification, phrases interrogatives, des fois style impérative.

Tableau 1.1 Indices textuels relatifs aux style objectif\ subjectif **[3]**

3.3. La polarité et l'intensité de l'opinion

La polarité peut être définie par des catégories telles que « positif », « neutre », et « négatif » La polarité d'une opinion exprime la positivité, la négativité ou une information de cette dernière. On dit d'une opinion positive qu'elle possède une polarité positive, et inversement, on dit d'une opinion négative qu'elle possède une polarité négative ou neutre possède une information.

L'intensité décrit à quel point la polarité d'une opinion est forte. Par exemple, dans une opinion à polarité positive, aimer est plus intense qu'apprécier, ou encore, dans une opinion de polarité négative, haïr est plus intense que de ne pas aimer.

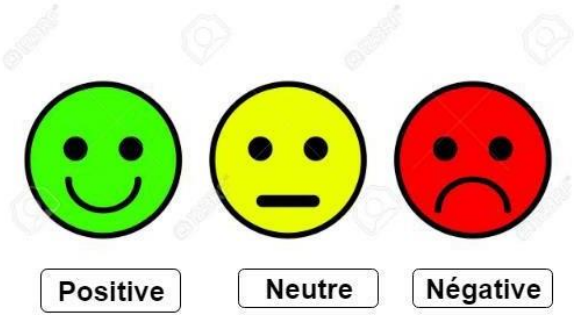


Fig.1.2 Représentation des différentes catégories d'opinions

4. Analyse des Sentiment (AS)

L'analyse des sentiments (Sentiment Analysis S.A) ou encore fouille d'opinion (Opinion Mining O.M) sont des termes utilisés pour représenter le processus de l'extraction automatiquement de l'orientation du sentiment ou de la polarité d'un avis sur un objet spécifique. Cet objet peut être une personne, un produit, un service, un événement, etc. En d'autres termes, elle détermine si une phrase ou un document est positif ou négatif ou neutre. Ces opinions sont exprimées en diverses formes telles que des articles, des commentaires, des forums, des messages, de courts commentaires, des tweets **[4]**.

Ce domaine émerge du T.A.L.N (Traitement Automatique du Langage Naturel).

Traitement Automatique du Langage Naturel

On regroupe sous le vocable de traitement automatique du langage naturel (TALN) l'ensemble des recherches et développements visant à modéliser et reproduire, à l'aide de machines, la capacité humaine à produire et à comprendre des énoncés linguistiques dans des buts de communication **[5]**.

4.1. Model formel de l'AS

Il existe plusieurs modèles formels qui définissent une opinion, le plus répondu est celui décrit dans les travaux de « Bing Liu » **[6]** et qui donne une représentation mathématique pour l'opinion comme suit :

L'opinion ou le sentiment peut être exprimé sur quoi que ce soit : un produit, un service, un sujet, un individu, une organisation, ou un événement. Généralement le terme « Objet » (*object*) est utilisé pour désigner l'entité qui a été commentée ou jugée. Un objet a un ensemble de composants (ou parties) et un ensemble d'attributs. Chaque composant peut également avoir ses sous- composants et l'ensemble de ses attributs, et ainsi de suite. Ainsi, l'objet peut être décomposé hiérarchiquement sur la base des parties de relation existantes entre ses composants.

***Objet :** Un objet O est une entité qui peut être un produit, un sujet, une personne, un événement ou une organisation. Il est associé à une paire, O: (T, A), où T est une hiérarchie de composants (taxonomie ou parties) et de sous-composantes de O et A est un ensemble d'attributs de O. Chaque composant O_i - appartenant à T - a son propre ensemble de sous- composants T_i et d'attributs A_i i.e. O_i: (T_i, A_i).

On peut représenter cette hiérarchie comme un arbre dont la racine est l'objet lui-même, chaque nœud non racine est soit un composant ou sous-composant de l'objet, chaque lien est une partie de relation, chaque nœud est associé à un ensemble d'attributs. Un avis peut être exprimé sur n'importe quel nœud et sur n'importe quel attribut d'un nœud.

Cependant, il est probablement trop complexe à utiliser une représentation hiérarchique, c'est pourquoi *Bing Liu* **[6]** propose un arbre aplati où le mot « caractéristiques » noté par F (*feature*) est utilisé pour représenter les composants et les attributs à la fois. L'utilisation du terme caractéristique pour un objet (en particulier un produit) est assez fréquente dans la pratique. On notera que dans cette définition, l'objet avec tous les nœuds de l'arbre sont des caractéristiques. On note par D un document évaluatif, qui peut être une revue de produit, un post sur un forum ou un blog qui évalue un objet particulier O. Dans le cas le plus général, D est composé d'une suite de phrases D = <S1, S2, ..., Sm>.

***Le passage d'opinion sur une caractéristique**

Le passage d'opinion sur une caractéristique **F** de l'objet **O** évalué en **D**, est un groupe de phrases consécutives dans **D** qui expriment un avis positif ou négatif sur **F**. Cela signifie qu'il est possible qu'une séquence de phrases (au moins une) exprime une opinion sur un objet **O** ou sur une de ses caractéristiques **F**, il est également possible qu'une seule phrase exprime des opinions sur plus d'une caractéristique **F**, par exemple « La qualité d'image de cette caméra est bonne, mais la durée de vie de la batterie est courte » ici on a deux caractéristiques **F1** et **F2** d'un même objet **O** qui sont évaluées, où : (**F1**= La qualité d'image. **F2** = La durée de vie de la batterie. **O** = La caméra)

***Le propriétaire de l'opinion (Opinion Holder)**

Le propriétaire ou l'émetteur d'une opinion donnée noté par **H** (opinion holder) est la personne ou l'organisation qui tient l'opinion donc qui a jugé une caractéristique donnée **F** d'un objet donné **O**. Dans le cas des critiques de produits ou des messages dans les forums et les blogs, les émetteurs d'opinion sont généralement les auteurs de messages. Les émetteurs d'opinion sont importants dans les articles de presse parce qu'ils affirment souvent explicitement la personne ou l'organisation qui a donné une opinion particulière. Par exemple, l'metteur de l'opinion **H** dans la phrase « ALI exprime son regret par rapport au boycott de Huawei par les états unis et ses alliés » est **H**=ALI.

F0 = O = Appareil photo numérique X :
F1 = Camera : Positive : 125 <examen individuel des phrases>
Négatif : 7 <examen individuel des phrases>
F2 = Qualité de l'image : Positive : 123 <examen individuel des phrases>
Négatif : 6 <examen individuel des phrases>
F3 = Taille : Positive : 82 <examen individuel des phrases>
Négatif : 10 <examen individuel des phrases>

Fig.1.3 Exemple d'application du modèle d'opinion de Bing Liu

Dans l'exemple représenté par la figure précédente l'objet O (racine de l'arbre) c'est l'appareil photo digital X et les chiffres exprimés pour chaque nœud correspondent aux nombres de phrases ayant une orientation sémantique favorable ou défavorable (+ ou -) pour chaque caractéristique Fi de O, sauf pour F0 où le calcul de l'orientation sémantique totale demande d'autre notions à mettre en jeu.

4.2. L'orientation sémantique

L'orientation sémantique (OS) d'un avis sur une caractéristique F d'un objet O indique si l'avis est positif, négatif ou neutre envers cette caractéristique. Le modèle de caractéristiques pour un objet donné et l'ensemble d'opinions sur ces caractéristiques peuvent définir un seul modèle qu'on appelle « modèle d'exploration de l'opinion basée sur les caractéristiques ».

4.3. Modèle d'exploration de l'opinion basée sur les caractéristiques

Un objet O est représenté par un ensemble fini de caractéristiques F = {f1, f2,..., fn}, qui comprend l'objet lui-même. Chaque caractéristique fi ∈ F peut être exprimée avec un ensemble fini de mots ou de phrases Wi, qui sont des synonymes, autrement dit, il y a un ensemble de synonyme correspondant W = {W1, W2, ..., Wn} pour les n caractéristiques. Dans un document d'évaluation D qui évalue l'objet O. Un propriétaire d'opinion (j) juge un sous-ensemble de caractéristiques Sj ⊆ F. Il donne son opinion sur chaque caractéristique fk∈ Sj en choisissant un mot ou une phrase à partir de Wk pour décrire la caractéristique fk, puis il en exprime son avis (positif, négatif ou neutre). La tâche de l'AS est de découvrir tous ces éléments cachés de l'information à partir d'un document d'évaluation donnée D.

Le résultat de l'AS pour un document d'évaluation D est un ensemble de quadruplets. Chaque quadruple est noté (H, O, F, OS), où H est le propriétaire d'opinion, O est l'objet, F est une caractéristique de O et OS est l'orientation sémantique de l'opinion exprimée sur la caractéristique F dans une phrase de D.

Les opinions neutres sont ignorées dans la sortie car ils ne sont généralement pas utiles. Compte tenu d'une collection de documents d'évaluation D contenant des avis sur un objet O, trois problèmes techniques peuvent être identifiés (il y a clairement plus) :

**Problème 1* : Comment extraire les caractéristiques d'objets qui ont été commentés dans chaque document d ∈ D ?

Problème 2 : Comment déterminer si les opinions sur les caractéristiques sont positives, négatives ou neutres ?

Problème 3 : Comment Regrouper les synonymes des caractéristiques (car chaque propriétaire d'opinion peut utiliser différents mots ou phrase pour exprimer la même caractéristique) ?

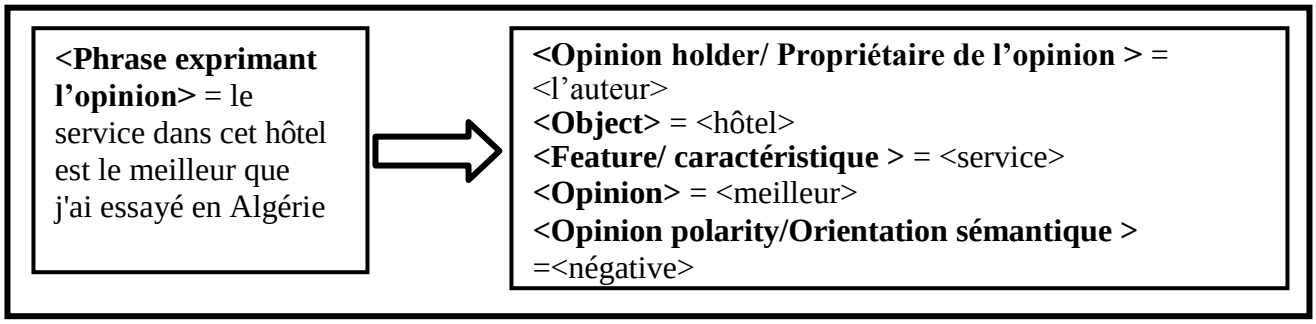


Fig.1.4 Exemple de la terminologie de l'analyse des sentiments

5. Tâches de l'Analyse des Sentiments

L'analyse des sentiments est une tâche complexe qui englobe plusieurs tâches distinctes, à savoir :

*Identification de l'objet de l'opinion

*Classification de la subjectivité (détection de la présence ou non de l'opinion)

*Classification du sentiment (positif, négatif, neutre).

*Classification de l'intensité de l'opinion (forte, moyenne, faible)

*Tâches complémentaires : extraction du porteur d'opinion, extraction d'entités et d'aspects

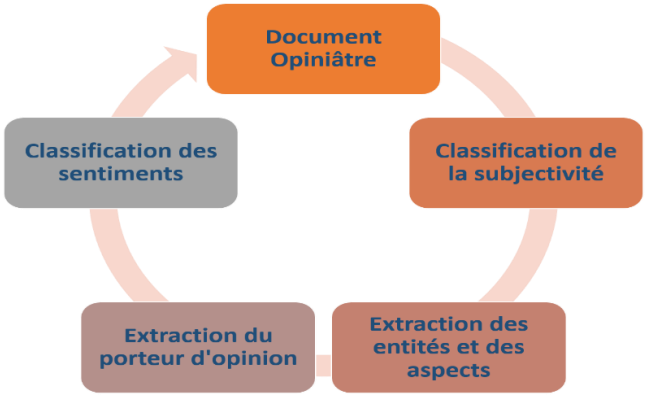


Fig.1.5 Taches de l'analyse des sentiments

5.1. Classifications de la subjectivité

En règle générale, un document donné contient des phrases exprimant une opinion et d'autres non. En d'autres termes, un document est un ensemble de phrases objectives (énonçant des faits) et subjectives (représentent l'opinion, le point de vue ou l'émotion de l'auteur). La classification de subjectivité consiste à classer les phrases en opinions ou en non-opinions. Tang et al **[7]** ont déclaré la classification de subjectivité suivante : S = {s1, . . .} Un ensemble de phrases dans le document D. Le problème de la classification de subjectivité est de distinguer les phrases utilisées pour présenter des opinions et d'autres formes de subjectivité (phrases subjectives S) des phrases utilisées pour présenter objectivement des informations factuelles (ensemble de phrases objectives So)., où Ss U So = S.

D'autres modèles proposent une solution plus générique en calculant le *pourcentage de subjectivité* pour tout le document, puis le traiter pour en extraire l'orientation sémantique en indiquant à la fin le pourcentage déjà calculé, ce qui donne plus d'information à l'intéressé. Dans cette orientation il y a lieu de mener une étude complète sur la reconnaissance de la subjectivité en utilisant différents indices et caractéristiques (la comparaison des résultats en utilisant les adjectifs, les adverbes et les verbes en prenant en compte la structure syntaxique comme par exemple l'emplacement des mots). L'exemple suivant illustre cette approche : la phrase " bad movie " signifiant "un mauvais filme" est à 80% négative avec un pourcentage de subjectivité égale à 90% : suivant l'analyseur « Sentiment Analysis with Python NLTK Text Classification »[2]

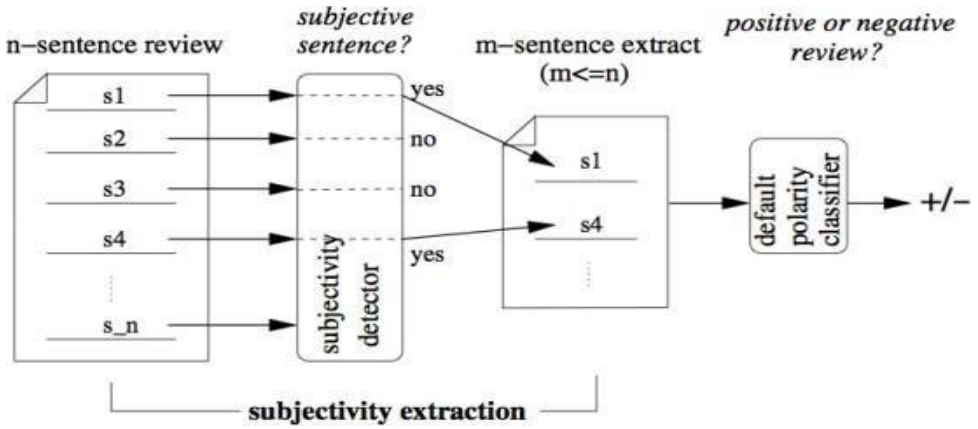


Fig.1.6 Processus de fouille d'opinion passant par une détection des phrases subjectives **[8]**

[2] Source : http://text-processing.com/demo/sentiment/ [visité le : 08/04/2019]

5.2. Classification des sentiments

Il y a des modèles qui se limitent à juger si le document est « positif », « négatif » ou «neutre », c.à.d. que le texte favorise le sujet en question, le défavorise ou il n'est ni pour ni contre. La classification peut être multi-classes (extrêmement négative, négative, neutre, positive ou extrêmement positive).Tandis qu'ils y en a d'autres modèles qui proposent d'aller plus loin dans la classification tel la proposition qui donne autres classes : (amour, joie, surprise, colère, tristesse, crainte) par exemple les phrases : je suis déçus, i am angry (Je suis fâché), انا سعيد (je suis heureux) راني عيان (je suis fatigué) qui expriment successivement la déception, la colère ,la joie et la fatigue.

5.3. Extraction du porteur d'opinion

L'analyse des sentiments implique aussi des tâches facultatives comme l'extraction du porteur d'opinion, à savoir la découverte des détenteurs ou des sources d'opinion. La détection du porteur d'opinion consiste à reconnaître les sources directes ou indirectes de l'opinion. Ils sont essentiels dans les articles de presse et autres documents officiels parce que plusieurs opinions peuvent être exprimées dans le même article correspondant à différents détenteurs (porteurs) d'opinion. Dans les documents de ce genre, les multiples détenteurs d'opinion peuvent être explicitement mentionnés par leurs noms. Dans les réseaux sociaux, les sites et blogs...etc. le porteur d'opinion est habituellement l'auteur qui peut être identifié par les identifiants de connexion.

5.4. Extraction d'entités et d'aspects

Est une tâche supplémentaire consiste en la découverte de l'entité cible. Les blogs et les sites de médias sociaux ont tendances à ne pas avoir un sujet prédéfini et sont par conséquent, enclins à discuter de différents sujets. Dans de telles plates-formes, il devient nécessaire de connaître l'entité cible. En outre, comme mentionné précédemment les entités cibles peuvent avoir différents aspects. Un internaute peut avoir des opinions divergentes sur les différents aspects de l'entité cible.

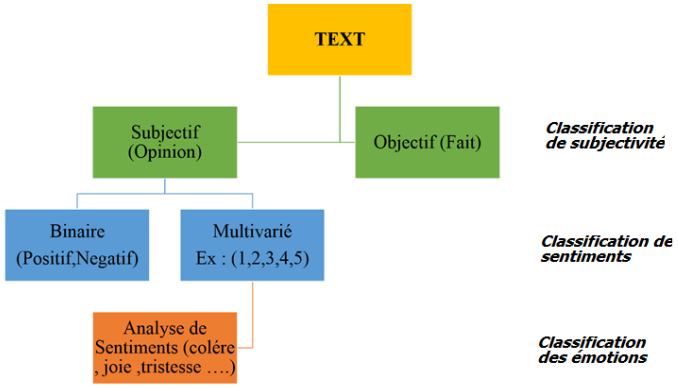


Fig.1.7 Niveaux de classification

6. Niveaux de l'Analyse des Sentiments

L'analyse des sentiments a été étudiée principalement sur trois niveaux

6.1. Niveau document

Il s'agit de donner une évaluation à propos d'un article ou un produit en assumant que le document ne traite qu'un seul sujet (un seul objet) mais cette méthode n'est pas applicable pour tous les documents car on peut avoir pas mal de documents qui traitent plusieurs sujets à la fois.

6.2. Niveau phrase

La tâche à ce niveau consiste en la détection des phrases subjectives dans un document à partir d'un mélange de phrases objectives et subjectives et ensuite, déterminer si chaque phrase a exprimé une opinion positive, négative ou neutre. Neutre signifie généralement « pas d'opinion ». Cette méthode reste toujours insatisfaisante dans plusieurs cas : supposant que la phrase contient plusieurs opinions différentes, donc on aura un problème d'existence de termes contradictoires, par exemple dans chacune des phrases suivantes il y a deux opinions différentes : « Le quartier était magnifique mais elle ne l'a pas aimé »

6.3. Niveau aspects

Les deux niveaux d'analyse de document et de phrase ne nous permettent pas exactement de découvrir ce que les gens apprécient et n'apprécient pas, le niveau d'aspect effectue une analyse plus fine. Au lieu de regarder les éléments de langage (documents, paragraphes, phrases), le niveau d'aspect regarde directement l'opinion elle-même. Il est basé sur l'idée que l'opinion se compose d'un sentiment positif ou négatif, et d'une cible d'opinion.

7. Approches de l'Analyse des Sentiments

Les approches de détection d'opinions cherchent à déterminer, de la manière la plus automatique possible, les caractéristiques d'opinions positives ou négatives. Enormément de travaux ont été effectués sur le sujet, et deux grandes catégories de méthodes peuvent être mises en avant : les approches linguistiques (basé lexique), les approches statistiques (basé corpus).

7.1. Approches basées sur le lexique

La méthode basée sur le lexique. Il utilise un lexique composé de termes avec des scores de sentiment respectifs pour chaque terme. Le terme peut être associé à un seul mot ou une phrase. Le sentiment est défini en fonction de la présence ou de l'absence de termes dans le lexique. L'approche basée sur le lexique inclut l'approche basée sur le corpus et

l'approche basée sur le dictionnaire.

7.1.1. Approche Basée Sur Le Dictionnaire

L'idée principale derrière l'approche basée sur un dictionnaire est d'utiliser des bases de données lexicales avec des mots d'opinion pour extraire le sentiment du document. La procédure de recherche est itérative. À chaque itération, l'algorithme prend un ensemble de mots mis à jour (ensemble étendu) et effectue une nouvelle recherche jusqu'à ce qu'il n'y ait plus de nouveaux mots à inclure. En fin de compte, un ensemble de mots de sentiment peut être examiné dans le but de supprimer les erreurs.

7.1.2. Approche Basée Sur Le Corpus

Dans Bing Liu **[9]** indique que l'approche basée sur le corpus peut être appliquée dans deux cas. Le premier cas est une identification des mots d'opinion et de leurs polarités dans le corpus de domaine en utilisant un ensemble donné de mots d'opinion. Le second cas concerne la construction d'un nouveau lexique dans un domaine particulier à partir d'un autre lexique utilisant un corpus de domaine. Les résultats suggèrent que même si les mots d'opinion dépendent du domaine, il peut arriver que le même mot ait une orientation opposée selon le contexte.

7.2. Approche D'apprentissage Automatique

Appelée aussi approche statistique, cette approche se basée sur l'apprentissage automatique. Elle utilise la technique de classification pour classer le texte en des classes déférentes. Il existe principalement deux types de techniques d'apprentissage.

7.2.1 Apprentissage non supervisé

Basé sur des entrées simples, sans aucune mention de cibles. Cela dépend simplement du regroupement.

7.2.2. Apprentissage supervisé

Définit les cibles prédéfinies qui doivent être atteintes, ainsi que les entres. Les ensembles de données sont formés pour atteindre des résultats significatifs lorsqu'ils sont rencontrés lors de la prise de décision

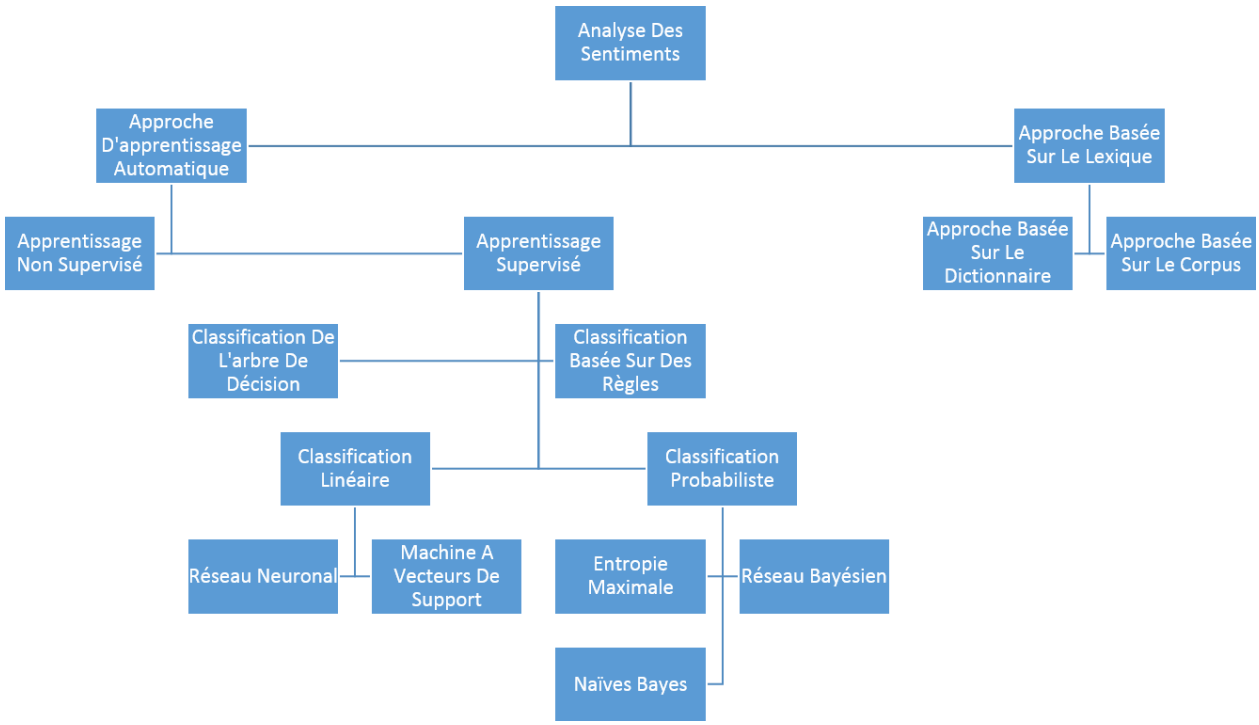


Fig.1.8 Les approches du A.S **[10]**

8. Domaines d'applications de l'Analyse des Sentiments

**Politique* : Grâce à l'analyse des sentiments, les décideurs de politique pouvant prenons l'avis des citoyens sur certaines politiques, afin de bénéficier de cette information pour améliorer ou créer une nouvelle politique qui convient avec les citoyens.

**Prise de décision :* L'opinion et l'expérience des gens sont un élément très utile dans le processus de prise de décision.

**Les systèmes de recommandations :* À travers l'analyse des sentiments en peut class les opinions des gens on positive ou négative, le system définir qui devrait prendre la recommandation et qui ne devrait pas prendre la recommandation.

**Domaine de Transport :* Pour assembler et analyses les opinions de public sur le statut de transport. Exemple : Système de transport intelligent moderne.

**Domaine médical :* Analyse opinion des médicine, patient sur les médicaments et les services hospitalier. Ainsi sur les documents de l'état de patient qui contient le diagnostic et la description du résultat d'examen.

**Domain éducation :* Développer le niveau d'enseignement à travers l'analyse et l'interprétation de l'opinion des étudiants à travers les méthodes d'enseignement ce qui permet d'améliorer l'enseignement et l'apprentissage.

**Marketing :* Du côté entreprises, permet au fournisseur plus de connaissances à propos

des besoins des consommateurs, du côté client il peut donner son opinion, s'inspirer des opinions d'autres clients pour l'aidé à sa décision et aussi comparer les produits avant de les acquérir.

**Economique* :la collecte et l'analyse des opinions des individus sont devenues des sources d'informations précieuses pour les fabricants car ils peuvent recueillir les opinions favorables et défavorables au sujet de leurs produits ou services et de ce fait ils peuvent améliorer la qualité de leurs produit sou services et ainsi augmenter leurs profits. La bourse (Prévision boursière), les acteurs des marchés financiers et les brokers aussi ont rapidement compris l'intérêt de l'analyse des sentiments. Des agences vendent aux entreprises la traque des moindres mots sur leur image, et sur leurs produits.

**Sociale :* bien que l'achat d'un produit ou d'un service, en prenant une bonne décision n'est plus une tâche aussi difficile. Mais par cette technique, les gens peuvent facilement évaluer les opinions et expériences des autres concernant n'importe quel produit ou service et ils peuvent aussi facilement comparer les marques concurrentes. Maintenant, les gens ne veulent pas se fier à un conseiller externe. L'analyse des sentiments extrait les opinions des gens à partir de l'immense collection de contenu non structuré, l'Internet, les analyse et les présente de façon très structurée et compréhensible.

8. Difficultés de l'analyse de sentiment

L'extraction du sentiment ou d'opinion consiste à déterminer la polarité d'une telle opinion. Dans ce qui suit nous citons quelques difficultés de cette procédure.

*Ambiguïté de certains mots positifs ou négatifs selon les contextes et qui ne peut pas toujours être levée.

*Difficulté due aux structures syntaxiques et sémantiques d'une phrase et l'expression de l'opinion. Par exemple " l'histoire du film est intéressante mais les acteurs étaient mauvais ". Dans ce cas la polarité de la deuxième partie est opposée à la première.

*Difficulté due au contexte : la nécessite d'une bonne analyse syntaxique du texte ; analyse qui peut se révéler particulièrement difficile dans des cas de coordination entre plusieurs parties d'une phrase. Par exemple "ma tonte a bien préparé le gâteau, son décor est bon mais je n'ai pas aimée le goût", l'opinion de la dernière partie de la phrase est la plus importante.

*Difficulté due à l'analyse de la phrase par " paquets de mots ". Les deux phrases suivantes contiennent les mêmes paquets de mots sans pour autant exprimer les mêmes sentiments. La première phrase contient un sentiment positif alors que la deuxième est négative : " Je l'ai apprécié pas seulement à cause de ...", " Je l'ai pas apprécié seulement à cause de ... " ou se présente la gestion de négation.

*Difficulté due au langage qu'utilisent les internautes pour s'exprimer. Les ponctuations ne sont pas forcément utilisées pour marquer les fins de phrases, des mots spécifiques sont utilisés tel que : «ha ha ha», «Goood», «super».

*Difficulté de déterminer un lexique adapté à l'analyse de l'ensemble des textes d'opinion.

9. Les défis d'analyse les sentiments

*Le langage et sa structure constituent le principal défi. Cependant, ils existent plusieurs problèmes et défis dans le domaine de l'analyse des sentiments.

*L'état émotionnel du locuteur : Les sentiments du locuteur peuvent être compatibles ou contraires aux déclarations faites par le locuteur.

*Succès ou échec d'un côté avec respect à un autre : Exemple 'Yay! France beat Germany 3–1': si en a un supporteur de la France émotion positif, si en a un supporteur de l'Allemagne émotion négative.

*Déclaration neutre des informations validées : Si en a aucune indication sur l'motion mais en a des citations valides, donc on ne sait pas et ce qu'en a des déclarations neutres ou une déclaration négatif d'émotion.

*Sarcasme : Est de déclarer des sentiments positifs, même s'ils sont négatifs. Exemple : « croyez-vous vraiment ce que vous dites ? »

*Différent sentiment vers différent cible d'opinion : L'orateur peut exprimer l'opinion à propos de cibles multiples, et le sentiment envers les différentes cibles pourrait être différent.

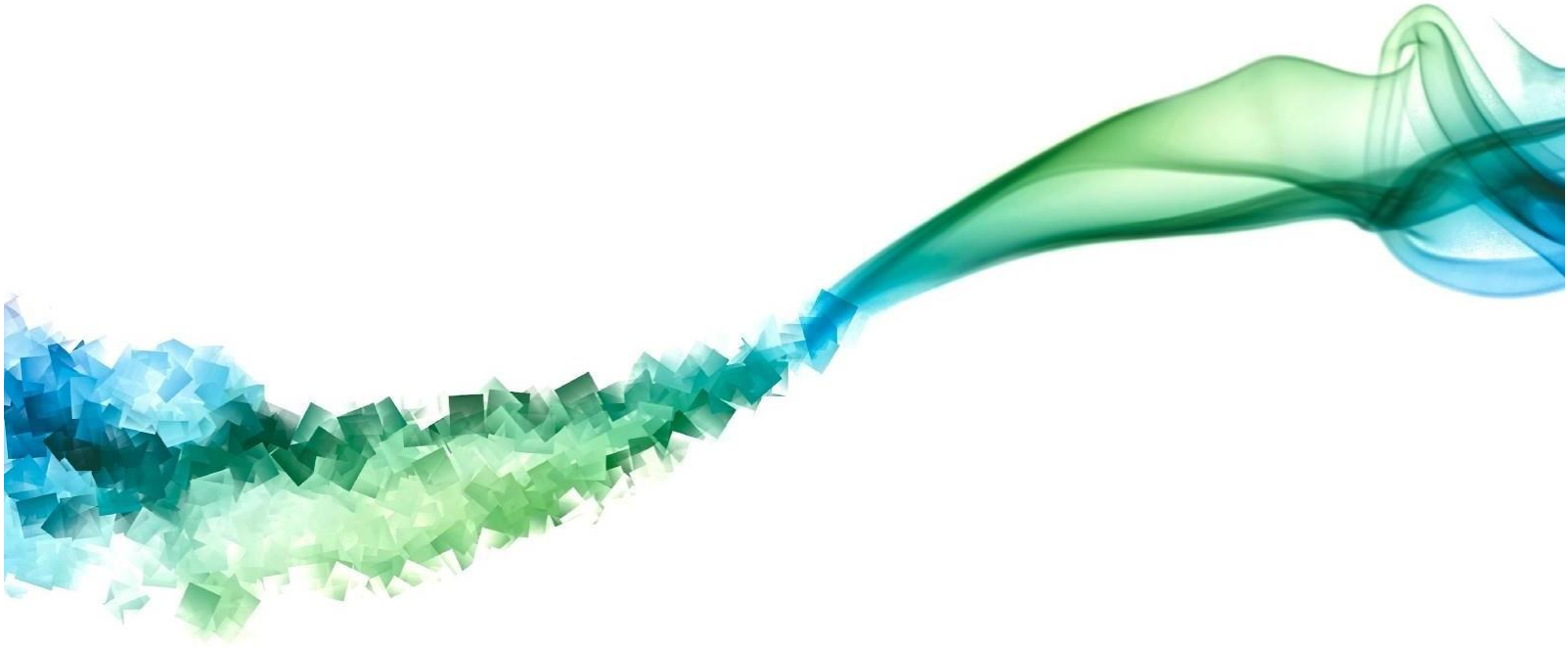
*Déterminer précisément la cible de l'opinion : Parfois, il est difficile de préciser identifier la cible d'opinion, par exemple, sur un sujet relatif aux conditions de travail ou aux prestations du restaurant, c'est le salarié concerné qui sera interrogé.

10. Conclusion

A l'heure où le *big-data* représente l'un des grands défis technologiques et économiques actuels, de nombreux systèmes d'analyse ont été développés afin d'offrir une meilleure exploitation de ces données massives ; parmi ces systèmes, l'analyse de sentiments, qui a été développés afin d'identifier rapidement les sentiments clés issus d'une importante quantité de données textuelles disponible généralement sur le web.

Dans ce chapitre on a introduit le domaine de l'analyse des sentiments : définition, caractéristiques, techniques, applications et défis.

Chapitre 02



Word-Embedding pour l'Arabe

1. Introduction

Dans le but de la réalisation d'un système d'analyse de sentiment fiable on va avoir besoin d'un corpus lexical. L'idée, est qu'il soit équilibré par catégorie et relativement consistant, cette première étape est essentielle afin de s'assurer du bon fonctionnement du reste des traitements.

La seconde étape sera de faire soumettre nos données non-structurées de texte à un traitement spécifique pour en sortir des caractéristiques (*features)* utilisables et structurées, par exemple sous forme vectorielle.

Le choix d'une bonne représentation des mots est indispensable, une fois notre corpus formaté, on va pouvoir appliquer les méthodes (Réseaux de Neurones) sur ces données afin de réaliser notre model AS.

On appelle la technique de représentation d'un mot, ou un ensemble de mots en vecteurs de dimension inférieure : Word embedding, c'est l'une des représentations les plus populaires du vocabulaire de document, parmi ses modèles le Word2vec (CBOW, Skip-Gram), GloVe et FastText.

Bien que l'analyse des sentiments soit un axe de recherche prometteur, les études réalisées pour la langue arabe sont encore très limités et ce dû au manque des travaux effectués notamment ceux relatifs à la création des lexiques correspondants.

Le présent chapitre est divisé en trois parties, la première : un aperçu sur la représentation vectorielle et les concepts relatifs, la deuxième : description de la langue arabe et ses caractéristiques d'un point de vu informatique, la troisième : nous mettons l'accent sur quelques études liées à l'application des modèles du Word embedding pour la langue arabe.

2. La représentation vectorielle

2.1. Le Deep Learning

2.1.1. Définition

Deep Learning autrement dit Apprentissage profond : est une forme d'intelligence artificielle dérivé du *machine Learning* (apprentissage automatique) où la machine est capable d'apprendre par elle-même, contrairement à la programmation où elle se contente d'exécuter à la lettre des règles prédéterminées **[11]**,le domaine du D.L traite des algorithmes inspirés de la structure et de la fonction du cerveau, appelés réseaux de neurones artificiels. En d'autres termes, il reflète le fonctionnement de notre cerveau. Ces algorithmes s'apparentent à la façon dont le système nerveux est structuré, où chaque

neurone est connecté et transmet des informations, le D.L a été introduit pour la première fois dans par les travaux de *Dechter.R* et *al* **[12].**

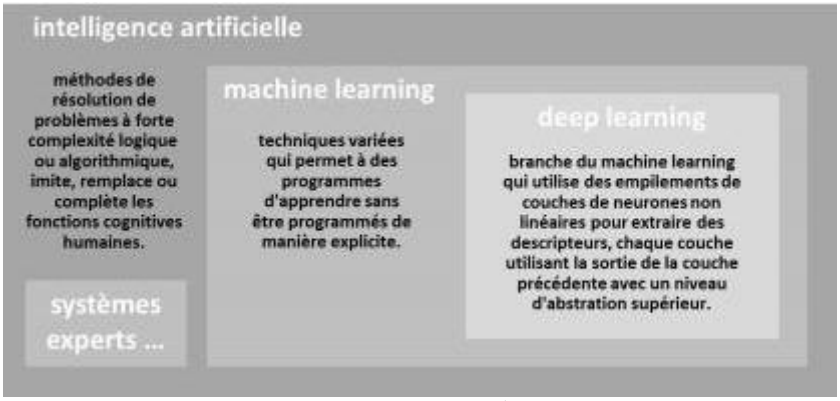


Fig. 2.1 Relation IA/ML/DL[1]

2.1.2. Fonctionnement du Deep Learning

Le DL s'appuie sur un réseau de neurones artificiels s'inspirant du cerveau humain. Ce réseau est composé de dizaines voire de centaines de « couches » de neurones, chacune recevant et interprétant les informations de la couche précédente. Le système apprendra par exemple à reconnaître les lettres avant de s'attaquer aux mots dans un texte, ou détermine s'il y a un visage sur une photo avant de découvrir de quelle personne il s'agit **[13].**

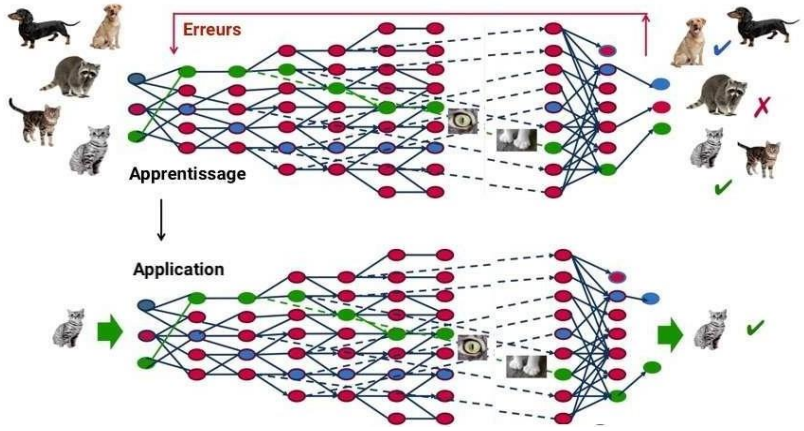


Fig. 2.2 Fonctionnement du Deep Learning[2]

« À travers un processus d'autoapprentissage, le deep Learning est capable d'identifier un chat sur une photo. À chaque couche du réseau neuronal correspond un aspect particulier de l'image »

[1] Source Photo © Google / DeepMind
[2] Source Photo © Google /MapR, C.D, Futura

2.1.3. Domaines d'application

La plupart des acteurs du domaine ne jurent aujourd'hui plus que par le D.L. Google, Apple, Facebook…, Apple et Microsoft mettent d'ailleurs tous leur propre librairie de DL à la disposition des développeurs.

Le D.L est utilisé dans de nombreux domaines :(Reconnaissance d'image, Traduction automatique, Voiture autonome, Diagnostic médical, Recommandations personnalisées ,Modération automatique des réseaux sociaux, Prédiction financière et trading automatisé, Identification de pièces défectueuses, Détection de malwares ou de fraudes, Chatbots , Exploration spatiale, Robot intelligents ……

Dans le contexte de notre travail le D.L a renouvelé les perspectives de recherche en traitement automatique des langues naturelles (T.A.L.N) dont la plupart des applications (analyse syntaxique et sémantique des textes et du discours, résumé et traduction automatique, ...) nécessitent de modéliser des données structurées qui se caractérisent par des distributions particulières, parcimonieuses et avec des espaces de réalisations de grande dimension. Le D.L a permis des avancées importantes en ce qui concerne les représentations et le traitement.

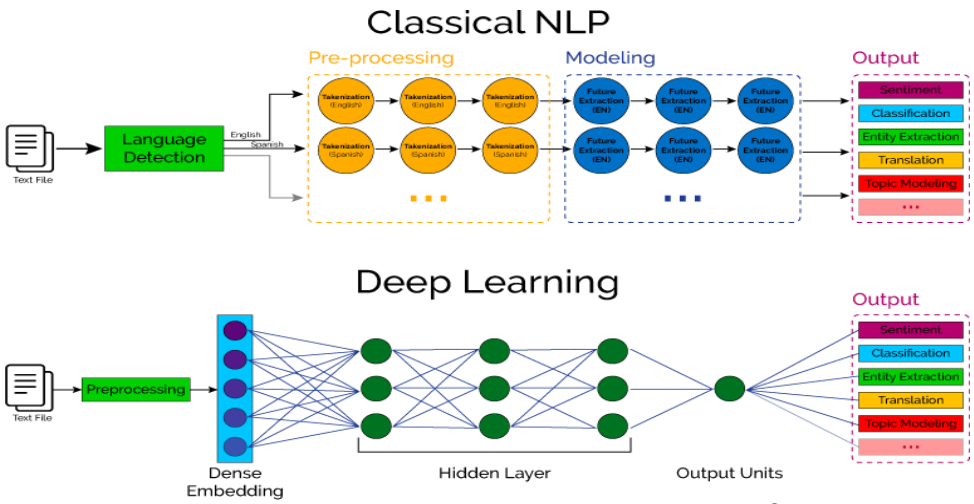


Fig. 2.3 Apport du DL dans le T.A.L.N [3]

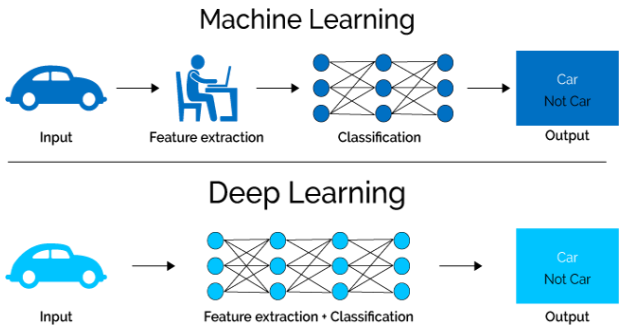


Fig. 2.4 La différence de fonctionnement entre DL et ML (Google Trends)

[3] https://www.pinterest.com/pin/

3. Word-Embedding

3.1. Définition

Le Word Embedding **[14]** (en français l'incorporation de mots ou plongement de mots) est une méthode d'apprentissage d'une représentation de mots utilisée notamment en traitement automatique des langues.

Cette technique permet de représenter chaque mot d'un dictionnaire par un vecteur de nombres réels correspondant. Ceci facilite l'analyse sémantique des mots. Cette nouvelle représentation a ceci de particulier que les mots apparaissant dans des contextes similaires possèdent des vecteurs correspondants qui sont relativement proches. Par exemple, on pourrait s'attendre à ce que les mots « chien » et « chat » soient représentés par des vecteurs relativement peu distants dans l'espace vectoriel où sont définis ces vecteurs.

Le W.E constituent une méthode pour mitiger un problème récurrent en intelligence artificielle, soit celui du fléau de la dimension . En effet, sans les plongements de mots, les objets mathématiques utilisés pour représenter les mots ont typiquement un grand nombre de dimensions, tant et si bien que ces objets se retrouvent « isolés » et deviennent épars. La technique des Word embeddings diminue le nombre de ces dimensions, facilitant ainsi les tâches d'apprentissage impliquant ces mots. Il existe différents modèles d'intégration de mots, tels que Word2vec (Google), GloVe (Stanford) **[15]**.

Le WE est également appelée modèle sémantique distribué ou modèle vectoriel d'espace sémantique ou d'espace représenté vectoriel distribué. En lisant ces noms, on rencontre le mot sémantique, qui signifie catégoriser des mots similaires. Par exemple, les fruits comme les pommes, les mangues et les bananes doivent être placés à proximité, tandis que les livres seront loin de ces mots. Dans un sens plus large, l'incorporation de mots créera le vecteur de fruits qui sera placé loin de la représentation vectorielle des livres**.**

1. I enjoy flying.
2. I like NLP.
3. I like deep learning.

The resulting counts matrix will then be:

$$X = \begin{array}{c} \\ I \\ like \\ enjoy \\ deep \\ learning \\ NLP \\ flying \\ . \end{array} \begin{array}{c} \begin{array}{cccccccc} I & like & enjoy & deep & learning & NLP & flying & . \end{array} \\ \begin{bmatrix} 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Fig. 2.5 Exemple de représentation vectorielle [4]

[4] Source photo https://towardsdatascience.com/word-to-vectors-natural-language-processing

3.2. Principe général du WE

Le WE est basé sur la sémantique distributionnelle « Des mots apparaissant dans contextes similaires ont des sens proches » **[16]** "You shall know a word by the company it keep" (Vous connaîtrez un mot par ses fréquentations)

L'idée est que des mots de sens proches auront tendance à apparaître dans des voisinages de mots similaires. En utilisant le voisinage d'un mot pour représenter ce dernier, on crée ainsi une représentation qui peut encoder sa sémantique (son sens). En d'autres termes, le contexte suffit pour représenter un mot **[17]**.

3.3. Pourquoi le World Embedding

*Le W.E aide à la création d'entités, au regroupement de documents, à la classification de texte et aux tâches de traitement de langage naturel, a beaucoup d'application :

*Calcul des mots similaires : l'incorporation de mots est utilisée pour suggérer des mots similaires au mot soumis au modèle de prédiction. Parallèlement à cela, il suggère également des mots dissemblables, ainsi que les mots les plus courants.

*Création d'un groupe de mots apparentés : il est utilisé pour le groupement sémantique qui regroupe des éléments de caractéristiques similaires et dissemblables au loin.

*Fonctionnalité pour la classification du texte : le texte est mappé dans des tableaux de vecteurs qui alimentent le modèle pour la formation et la prédiction. Les modèles de classificateur basés sur le texte ne peuvent pas être formés sur la chaîne, ce qui convertira le texte en une forme pouvant être entraînée par une machine. En outre ses fonctionnalités de construction d'aide sémantique dans la classification basée sur le texte.

*Le regroupement de documents est une autre application où l'incorporation de mots est largement utilisée, notamment dans la classification des textes.

*Traitement du langage naturel (TALN) : il existe de nombreuses applications où l'incorporation de mots est utile et l'emporte sur les phases d'extraction de fonctions, l'analyse des sentiments et l'analyse syntaxique, la traduction, la reconnaissance vocale…etc.

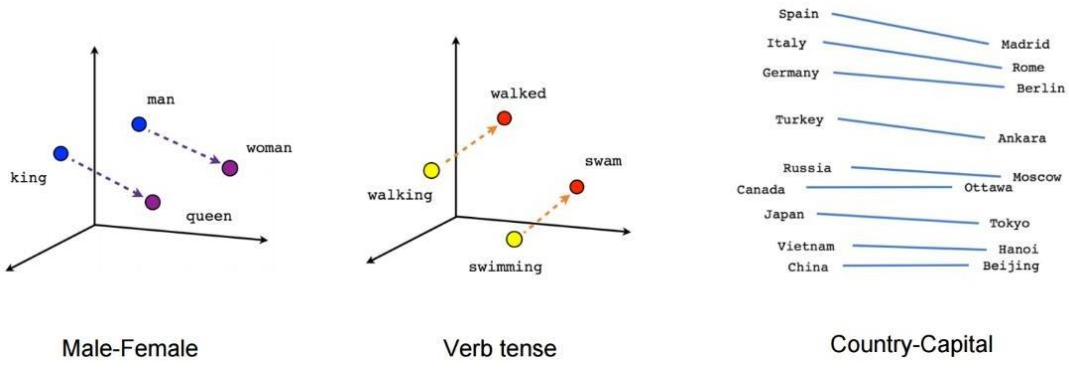


Fig.2.6 Les fonctionnalités de Word embedding (similarité, regroupement)[5]

4. Word2Vec

4.1. Définition

Word2Vec est un groupe de modèles associés utilisés pour produire des mots incorporés, techniquement le word2vec est dérivé du Word Embedding. Les modèles constituants sont des réseaux de neurones à deux couches peu profondes formés pour reconstruire les contextes linguistiques des mots. Word2Vec prend en entrée un grand corpus de texte et crée un espace vectoriel, généralement de plusieurs centaines de dimensions, chaque mot unique du corpus étant associé à un vecteur correspondant. Les vecteurs de mots sont positionnés dans l'espace vectoriel de telle sorte que les mots partageant des contextes communs dans le corpus soient représentés par des vecteurs numériques proches **[18]**.

Word2Vec est l'algorithme le plus connus de Word embedding . Il a été développé par une équipe de recherche de Google sous la direction de Tomas Mikolov.

4.2. Caractéristique du word2vec

Word2Vec possède différentes caractéristiques, dont les plus importants sont :

La dimensionnalité de l'espace vectoriel à construire, c'est à dire le nombre de descripteurs numériques utilisés pour décrire les mots (entre 100 et 1000 en général).

La taille du contexte d'un mot, c'est à dire le nombre de termes entourant le mot en question (les auteurs suggèrent d'utiliser des contextes de taille 10 avec l'architecture Skip- Gram et 5 avec l'architecture CBOW).

Étant donné que Word2Vec n'est composé que de deux couches, cet algorithme est rapide à entraîner et à exécuter, ce qui se révèle être un avantage important par rapport à d'autre méthodes de Word embedding.

[5] Source de photo site open class room https://openclassrooms.com/-analysez-vos-donnees-textuelles/

4.3. Pourquoi le word2vec

Word2Vec symbolise les mots dans la représentation de l'espace vectoriel. Les mots sont représentés sous forme de vecteurs et le placement est fait de manière à ce que des mots ayant le même sens apparaissent ensemble et que des mots dissemblables soient éloignés. Ceci est également appelé une relation sémantique. Les réseaux de neurones ne comprennent pas le texte mais ne comprennent que des nombres. Le word2vec fournit un moyen de convertir du texte en un vecteur numérique

Les applications du word2vec englobent ceux du W.E : calcul de la similarité, les mots proches sémantiquement, la non similarité, regroupement des mots apparentés, classification des textes, regroupement des textes, traitement du langage naturel (l'analyse sentimentale et d'analyse syntaxique, la traduction, la reconnaissance vocale…etc), par conséquence le word2vec peut être utile dans différents domaines :

*L'analyseur de dépendance utilise word2vec pour générer une relation de dépendance meilleure et plus précise entre les mots au moment de l'analyse.
*La reconnaissance d'entité nommée peut également utiliser word2vec, ce dernier étant très efficace pour découvrir une similarité dans la reconnaissance d'entité nommée (NER). Toutes les entités similaires peuvent se réunir et obtenir de meilleurs résultats.
*L'analyse des sentiments l'utilise pour préserver la similitude sémantique afin de générer de meilleurs résultats de sentiment. La similarité sémantique nous aide à savoir quel type de phrases ou de mots les gens utilisent pour exprimer leurs opinions, et on peut générer une bonne compréhension et une précision en utilisant les concepts de word2vec dans l'analyse des sentiments.
*La création d'une application qui prédit le nom d'une personne en utilisant son style d'écriture.
*Le classement des documents avec une grande précision et en utilisant des statistiques simples, on peut utiliser le concept word2vec pour classer les documents sans aucune étiquette humaine.
*Le regroupement de mots est le produit fondamental de word2vec. Tous les mots ayant une signification similaire sont regroupés.
*Google utilise word2vec et l'apprentissage en profondeur pour améliorer son produit de traduction automatique.

4.4. Les algorithmes de traitement

Word2vec a deux versions différentes. Ces versions sont les principaux algorithmes de word2vec : Continuous Bag of Words (CBOW) et Skip-Gram.

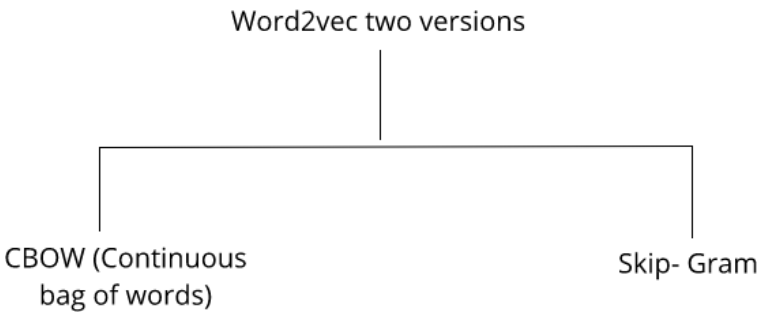


Fig.2.7 Versions de Word2Vec

L'apprentissage de la représentation des mots est essentiellement non supervisé, mais des cibles / étiquettes sont nécessaires pour former le modèle. Skip-gram et CBOW convertissent la représentation non supervisée en un formulaire supervisé pour l'entraînement.

Word2vec offre une option permettant de choisir entre CBOW et Skip-gram. Ces paramètres sont fournis lors de la formation du modèle.

4.4.1. Continuous Bag of Words CBOW

Dans CBOW, le mot actuel est prédit à l'aide de la fenêtre des fenêtres de contexte environnantes. Par exemple, si wi-1, wi-2, wi + 1, wi + 2 sont donnés mots ou contexte, ce modèle fournira wi.

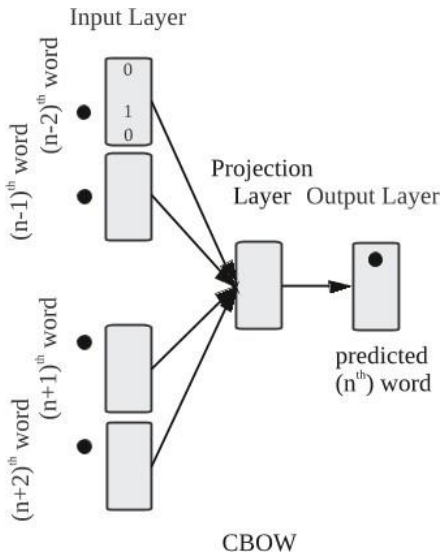


Fig. 2.8 Architecture CBOW [6]

[6] Crédit image: https://www.semanticscholar.org

Calculons les équations mathématiquement. Supposons que V soit la taille du vocabulaire et N la taille de la couche cachée. L'entrée est définie par {xi-1, xi-2, xi + 1, xi + 2}. Nous obtenons la matrice de poids en multipliant V * N. Une autre matrice est obtenue en multipliant le vecteur d'entrée par la matrice de poids. Ceci peut également être compris par l'équation suivante : h = xitW, où x it ∧ W sont respectivement le vecteur d'entrée et la matrice de pondération,
Pour calculer la correspondance entre le contexte et le mot suivant, on doit reporter à l'équation u = représentation prévue * h , où la représentation prédite est obtenue modèle ∧h dans l'équation ci-dessus.

4.4.2. Skip-Gram

Skip-Gram joue à l'opposé de CBOW, ce qui implique qu'il prédit la séquence ou le contexte donné à partir du mot. Vous pouvez inverser l'exemple pour le comprendre. Si wi est donné, cela prédira le contexte ou wi-1, wi-2, wi + 1, wi + 2.

Nous pouvons également en conclure que la cible est transmise à la couche d'entrée et que la couche de sortie est répliquée plusieurs fois pour tenir compte du nombre choisi de mots de contexte. Le vecteur d'erreur de toute la couche de sortie est additionné pour ajuster les poids via une méthode de rétro propagation.

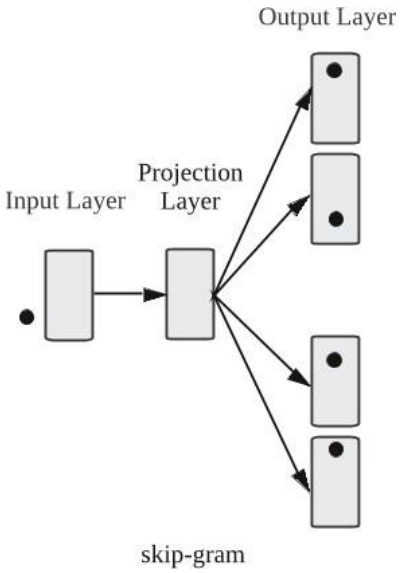


Fig.2.9 Architecture Skip gram [7]

4.4.3. Quel model à choisir

CBOW est plusieurs fois plus rapide que le Skip-Gram et fournit une meilleure fréquence

[7] Crédit image: https://www.semanticscholar.org

pour les mots fréquents, tandis que Skip-Gram nécessite une petite quantité de données d'entraînement et représente même des mots ou des phrases rares.

4.5. Évaluation de la qualité d'un modèle word2vec

Dans le but d'évaluer la qualité d'un modèle word2vec, ses créateurs (Mikolov et al) ont développé une approche **[19]** qui s'appuie sur les modèles sémantiques et syntaxiques. Ils ont développé un ensemble de 8 869 relations sémantiques et 10 675 relations syntaxiques qu'ils utilisent comme référence pour tester la précision d'un modèle. Lors de l'évaluation de la qualité d'un modèle vectoriel, un utilisateur peut s'inspirer de ce test de précision implémenté dans word2vec **[20]** ou développer son propre jeu de tests significatif pour les corpus constitutifs du modèle. Cette approche offre un test plus difficile que de simplement faire valoir que les mots les plus similaires à un mot de test donné sont intuitivement plausibles.

4.4. Systèmes inspirés/dérivés de word2vec

*Word2vecf **[21]** : intégration de dépendances syntaxiques dans les contextes utilises pour entrainer les embeddings.

*Dict2vec : utilisation de dictionnaires pour entrainer des embeddings.

***Projet Aravec 3.0 **[22]** : est un projet open source de représentation de mots distribués préformés (W.E) qui vise à fournir des modèles puissants pour le T.A.L.N arabe

*Nonce2vec : apprentissage de nouveaux mots sur très petits corpus.

*Corpus diachroniques

*Projet Evolex

Malgré que la technique de représentation proposée par le model Word embedding/ word2vec fonctionne avec n'importe quelle langue, mais la majorité des travaux sont en anglais, encore moins le français et l'espagnol, pour la langue arabe son utilisation reste très limité.

5. La langue arabe

5.1. Bref aperçu

La langue arabe comprend vingt-neuf lettres fondamentales (vingt-huit si l'on exclut la *hamza*, qui se comporte soit comme une lettre à part entière soit comme un diacritique). Il se lit et s'écrit de droite à gauche, comme beaucoup d'écritures sémitiques utilisant des abjads (syriaque, hébreu, etc).

De nombreuses lettres sont similaires par leur squelette (rasm) et ne se distinguent que par des points utilisés comme diacritiques au-dessus ou au-dessous de la ligne d'écriture (ـثـ ـنـ ـيـ ـبـ ـتـ). Il existe 18 formes de base (rasm). Les adaptations de l'alphabet arabe à d'autres langues se font sur ces mêmes formes de base, le plus souvent par l'ajout de points.

La plupart des lettres s'attachent entre elles, même en imprimerie, et leur graphie peut changer selon qu'elles sont en position initiale (liées à la lettre suivante mais pas la précédente), médiane (liées des deux côtés), finale (liée à la précédente mais pas la suivante) ou qu'elles sont isolées (sans liaison) : on parle de variantes contextuelles. La liaison peut être plus ou moins allongée sans changer la lecture des lettres : (كـتـب) (ktb), normalement compacté, peut également être rendu كـتـب en allongeant les liaisons, par exemple pour créer un effet calligraphique, ou pour des raisons de justification de mise en page.

Par ailleurs, six lettres (ا د ذ ر ز و) ne s'attachent jamais à la lettre suivante, de sorte qu'un mot peut être entrecoupé d'une ou plusieurs espaces.

L'alphabet arabe étant un *abjad*, le lecteur doit connaître la structure de la langue pour restituer les voyelles. Dans le cas de l'arabe, les voyelles d'un mot se répartissent au sein de la racine consonantique, suivant les règles de grammaire.

Dans les éditions du Coran ou les ouvrages didactiques, cependant, on utilise une notation vocalique plus ou moins précise sous forme de diacritiques. Il existe, de plus, dans de tels textes dits « vocalisés », une série d'autres diacritiques de syllabation dont les plus courants sont l'indication de l'absence de voyelle *(sukūn)* et la gémination des consonnes *(šadda)*.

Tous les mots en arabe sont dérivés d'une racine qui est composée de constantes. Ce sont généralement trois ou quatre lettres appelées radicaux.

La langue arabe est une langue riche morphologiquement « *Morphologically Rich Languages* (MRL) », en effet, une langue MRL est une langue dans laquelle les informations importantes concernant les unités syntaxiques et les relations sont exprimées au niveau du mot.

5.2. Caractéristique de la langue Arabe

L'Arabe est l'une des principales langues parlées dans le monde, c'est la langue maternelle de plus de 300 millions personnes. Elle est aussi majoritairement utilisée en d'autres pays non- arabes comme une deuxième langue officielle. En 1974, l'Arabe a été adopté comme l'une des six langues officielles des Nations Unies. Elle est aussi classée au quatrième rang parmi les langues les plus utilisées sur le Net en 2017, pourtant qu'elle n'a pas pu dépasser la septième place en 2010.

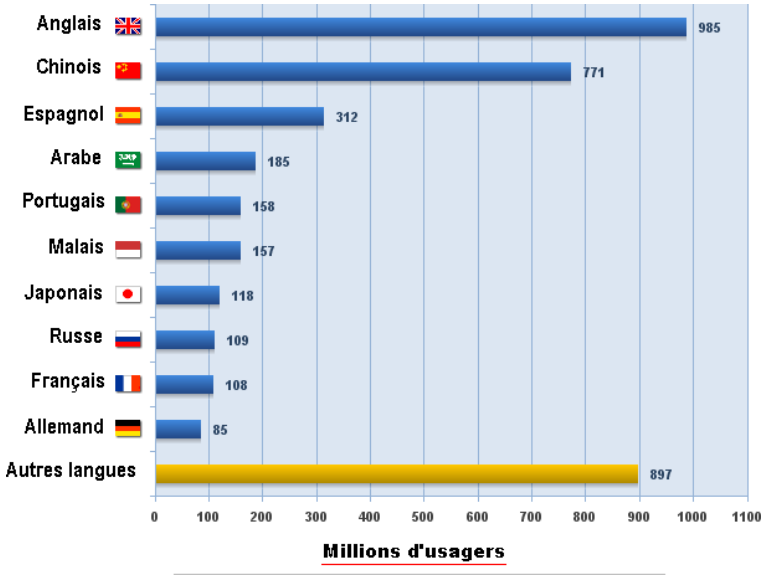


Fig.2.10 Les 10 langues plus utilisés sur internet en 2017[8]

L'Arabe est l'une des rares langues qui ont persisté pendant des siècles aux différents changements politiques et géographiques, elle s'est imposée avec la révélation coranique qui lui a conféré son statut de langue sacrée, elle est caractérisée par : un alphabet de 28 lettres, une orientation d'écriture de droite à gauche, un changement de forme des caractères selon leurs positions dans le mot. Morphologiquement, la majorité des mots Arabes sont dérivés à partir d'une liste de racine de 3 lettres (" ف", " ع" et " ل") qu'on appelle souvent les *patterns* de la langue Arabe, ces mots sont pratiquement devisés en trois classes :

Première classe : elle regroupe les noms ("مكتبة" qui veut dire "bibliothèque","كراس" qui veut dire "cahier"…), les adjectifs et les adverbes ("جميل" qui signifie "joli", "مطلقا" qui veut dire "absolument"…).

Deuxième classe : les verbes ("كتب" qui veut dire "écrire").

Troisième classe : les particules (prépositions, conjonctions de coordination…), par exemple : "في" signifiant "dans", "و" qui veut dire"et …

Dans le contexte des propriétés morphologiques et syntaxiques **[23]** on peut citer quelques-uns :

- La nature agglutinante de la langue : l'ensemble des morphèmes collés à l'unité lexicale1 véhiculent plusieurs informations morphosyntaxiques.
- La richesse flexionnelle de l'arabe
- L'absence de voyellation de la majorité des textes arabes écrits : ce phénomène entraîne un nombre important d'ambiguïtés morphologiques. En arabe, chaque lettre doit prendre

[8] Source Internet World State Minwatts Marketing Groupe

un signe de voyellation et de surcroît les voyelles finales sont porteuses de certains traits morphosyntaxiques comme la déclinaison, le mode, le cas.

- En outre des propriétés linguistiques, l'arabe recense un nombre de ressources linguistiques comprenant des lexiques monolingues et multilingues ainsi que des corpus de langue générale et des corpus de spécialité consacrés à une situation de communication ou à un domaine de la connaissance. L'arabe compte aussi un certain nombre d'outils linguistiques à savoir les analyseurs morphologiques ainsi que les racineurs basés essentiellement sur une procédure de dé suffixation qui consiste à supprimer les suffixes qui différencient les flexions des unités lexicales (les formes conjuguées d'un verbe par exemple).

5.3. La langue Arabe et ses variantes

L'arabe est une langue parlée par plus de 200 millions de personnes. Elle est langue officielle d'au moins 22 pays. C'est aussi la langue de référence pour plus d'un milliard de musulmans. Comme son nom l'indique, la langue arabe est la langue parlée à l'origine par le peuple arabe. C'est une langue sémitique (comme l'hébreu, l'araméen et le syriaque), son développement s'est accompagné d'une rapide et profonde évolution (en particulier dans la syntaxe et l'enrichissement lexical). L'arabe peut être considérée comme un terme générique rassemblant plusieurs variétés **[24]** :

-L'arabe classique : La langue du Coran, parlée au VIIe siècle ;

-L'arabe standard moderne (ASM) : Une forme un peu différenciée de l'arabe classique, et qui constitue la langue écrite de tous les pays arabophones. L'ASM reste le langage de la presse, de la littérature et de la correspondance formelle, alors que l'arabe classique appartient au domaine religieux et est pratiqué par les membres du clergé ;

-Les dialectes arabes : malgré l'existence d'une langue commune, chaque pays a développé son propre dialecte. Issus de l'arabe classique, leurs systèmes grammaticaux respectifs affichent de nettes divergences avec celui de l'ASM. On peut regrouper ces dialectes en quatre grands groupes :

1.les dialectes arabes, parlés dans la Péninsule Arabique (dialectes du Golfe, dialecte du najd, yéménite) ;
2.les dialectes maghrébins (algérien, marocain, tunisien, hassaniya de Mauritanie) ;
3.les dialectes proche-orientaux (égyptien, soudanais, syro-libano-palestinien, irakien) ;
4.la langue maltaise est également considérée comme un dialecte arabe.

5.4. Difficulté du traitement de la langue Arabe

Le retard de pénétration d'Internet dans les pays arabes n'est pas lié seulement aux

problèmes socio-économiques mais il dépend aussi des caractéristiques de la langue Arabe (cités un peu plus haut). Avant l'apparition d'Internet le code ASCII était le codage universel utilisé dans le monde qui ne prend en considération que les caractères Latins Anglais (de "A" jusqu'à "Z"), ainsi les utilisateurs et les développeurs parlant d'autres langues souffraient de beaucoup problèmes techniques dans les programmes utilisés comme les systèmes de traitement du texte, et d'affichage.

5.4.1. Codage des caractères Arabes

Les caractéristiques de la langue Arabe exigent des traitements particuliers afin d'être représentée dans des programmes informatiques. Le standard de codage ASCII a montré ses insuffisances dès sa première apparition, puisqu'il utilise juste sur 8 bits pour représenter un caractère. Au milieu des années 80, ISO (International Standard Organisation) déclenche l'idée d'utiliser un système universel où chaque caractère est codé sur plusieurs octets (de un à quatre) permettant à toutes les langues du monde d'être codée sur machine, ce qui a donné naissance aux fameux standards "Unicode" et "UTF_8" **[25]**.

5.4.2. Affichage du texte Arabe

Le mot est constitué d'une séquence de lettres, à l'affichage elles deviennent une suite de glyphes (représentation visuelle d'un caractère sur un dispositif d'affichage) **[26]**, pour la langue arabe le glyphe de chaque caractère n'est pas unique comme le cas des autres langues, il dépend de l'endroit où il se trouve dans la séquence (au début, au centre ou à la fin du mot), par exemple la lettre "س" (s) peut être représentée par trois glyphes différents ("ســ","ـســـ","ـس"), Ainsi les éditeurs du texte supportant l'Arabe doivent prendre en considération l'affichage du bon glyphe à la bonne position.

5.4.3 Les diacritiques :

L'écriture arabe courante ne note pas les voyelles, qui peuvent cependant apparaître sous forme de diacritiques dans certains textes à caractère didactique (Coran, apprentissage de la lecture, dictionnaires). De ce fait, un mot écrit en arabe peut généralement admettre plusieurs lectures suivant la répartition (ou l'absence) de voyelles et de redoublement de consonne, et s'apparente souvent à une sténographie : il faut pouvoir lire correctement un texte pour le comprendre, et il faut comprendre un texte pour le lire correctement.

5.4.4. Structure d'un mot :

En arabe un mot peut signifier toute une phrase grâce à sa structure composée qui est une agglutination d'éléments de la grammaire, la représentation suivante schématise une structure possible d'un mot.

**Proclitiques* sont des prépositions ou des conjonctions.

**Cors schématique* représente la forme de base pour chaque mot.

**Préfixes et suffixes* expriment les traits grammaticaux et indiquent les fonctions : cas du nom, mode du verbe et les modalités (nombre, genre, personne,)
**Enclitiques* sont des pronoms personnels.

Par exemple : le mot (أتعلمينهم) qui veut dire est ce que tu peux les enseigner ?

Enclitique	suffixes	Cors schématique	Préfixes	Proclitiques
هم	ين	علم	ت	أ
Objet masculin pluriel	Sujet féminin singulier	La racine	Préfixe verbale du temps	Conjonction d'interrogation

Tableau 2.2 Structure d'un mot arabe.

5.5. Traitement automatique de la langue arabe :

La langue arabe rencontre quatre principaux problèmes en traitement automatique : la segmentation du texte, l'agglutination des mots et détection de racine, l'absence de voyelles à l'écrit et l'étiquetage grammatical. Pour chacun de ces problèmes, tout système de traitement automatique doit traiter et enlever certaine ambiguïté.

5.5.1. Segmentation :

La segmentation d'un texte arabe est une étape fondamentale pour son traitement automatique ; son rôle est de découper le texte en unités d'un certain type qu'on aura défini et repéré préalablement. En effet, l'opération de segmentation d'un texte consiste à délimiter les segments de ses éléments de base qui sont les caractères, en éléments constituants différents niveaux structurels tels que : paragraphe, phrase, syntagme, mot graphique, mot-forme, morphème, etc.

Toutefois, les particularités de la langue arabe, rendent la segmentation arabe toujours différente, il n'y a pas de majuscules qui marquent le début d'une nouvelle phrase. De plus, les signes de ponctuation, ne sont pas utilisés de façon régulière.

La reconnaissance de la fin de phrase est délicate car la ponctuation n'est pas systématique et parfois les particules délimitent les phrases.

Pour la segmentation de texte utilise :

- Une segmentation morphologique basée sur la ponctuation,
- Une segmentation basée sur la reconnaissance de marqueurs morphosyntaxiques ou des mots fonctionnels comme : fonctionnels comme : حتى ، لكن ، أي ، و ، أو *ou, et, c.à.d., mais, quand.*

Cependant, ces particules peuvent jouer un autre rôle que celui de séparer les phrases**[27].**

5.5.2. Agglutination des mots et Détection de racine

La plupart des mots arabes sont composés par agglutination d'éléments lexicaux de base (proclitique + base + enclitique). Par exemple, la détermination peut s'exprimer par agglutination de l'article ال/al/ avant le mot (المالية/almaleya/ financement) ou par agglutination d'un pronom personnel après celui-ci (ماله/malohu/son argent) **[28]**.

Dans toute perspective de traitement automatique, le problème est donc de décomposer le mot en ces différentes parties. Cette décomposition nécessite des connaissances de niveau supérieur en cas d'ambiguïtés (si le mot accepte plusieurs segmentations).

Pour détecter la racine d'un mot, il faut connaître le schème par lequel il a été dérivé et supprimer les éléments flexionnels (antéfixes, préfixes, suffixes, post fixes) qui ont été ajoutés.

Nous utilisons une liste prédéfinie de préfixes et de suffixes **[29],** pour la lemmatisation de mots arabes ; ils ont été déterminés par un calcul de fréquence sur une collection d'articles arabes de l'Agence France Press (AFP) **[30].**

5.5.3. L'analyse sémantique

L'analyse sémantique tente de découvrir de façon plus générale le sens des mots, des phrases ou des textes entiers. C'est la phase la plus laborieuse pour les machines, et pour cette raison elle reste encore assez peu employée.

L'absence de voyelles peut générer des défauts de sens dans le traitement automatique, par exemple, le mot (العلم) isolé peut avoir plusieurs interprétations (*la science* ou *drapeau*) alors que voyellé sera (العِلْمُ pour *la science* et العَلَمُ pour *le drapeau*).

Les outils qui opèrent cette analyse font souvent appel à de gigantesques thésaurus (comme Arabic Wordnet pour l'arabe), permettant de classer chaque terme dans une arborescence de concepts pour déterminer les thèmes dominants d'un texte, ainsi qu'à des algorithmes complexes permettant d'évaluer les relations entre les différentes idées d'un texte donné.

5.5.4. Racineur

Les racineurs se veulent d'abord un outil utile au T.A.L.N, ce type d'analyse « simpliste », traite de façon identique affixes flexionnels et dérivationnels **[31]**. Les algorithmes de racinisation en arabe les plus connus sont :

- ***Racineur de larkey***

L'approche de *larkey* est une analyse morphologique assouplie. Elle consiste à essayer de déceler les préfixes et les suffixes ajoutés à l'unité lexicale : par exemple

le duel (ان) dans (معلمان , deux professeurs), le pluriel des noms masculins (ين , ون) dans (معلمون , des professeurs) et féminins (ات) dans (مسلمات , musulmanes) ; la forme possessive (نا , كم ,هم) dans (كتابهم , ses livres) **[32].**

- ***Racineur de Khoja***

Le racine de *Shereen khoja* développé au sein de l'université de Lancaster, a été utilisé dans le cadre d'un système de recherche d'information développé à l'Université du Massachusetts. L'approche de *Khoja* consiste à détecter la racine d'une unité lexicale, d'une part, il faut connaître le schème par lequel elle a été dérivée et supprimer les éléments flexionnels (préfixes et suffixes) qui ont été ajoutés, d'autre part comparer la racine extraite avec une liste des racines préalablement conçue **[32].**

6. Les travaux réalisés pour le Word embedding en langue arabe

6.1. Le travail de Rami Al-Rfou et al (Polyglot) [33]:

Représentations Word Embedding distribuées pour la PNL multilingue. La représentations Word embeddings de 06 langues (parmi eux l'arabe) en utilisant leur correspondant Wikipédia. Le vocabulaire de chaque langue contiendra jusqu'à 100 000 mots.

La démonstration quantitative de l'utilité de ces représentations en les utilisant comme caractéristiques uniques pour la formation d'une partie de tagueur de parole (tag of speech) pour un sous-ensemble de ces langues.

Mise en œuvre la formation de ces modèles ont été rendus possibles par des contributions à Theano (bibliothèque d'apprentissage machine). Ces optimisations habilitent les chercheurs à produire des représentations sous différents paramètres ou pour différents corpus que Wikipédia.

Pour l*'Arabe* - Le premier exemple montre le mot "Merci". En dépit de ne pas enlever les diacritiques à partir du texte, le modèle a appris que les deux formes de surface du mot signifient similaire les choses et, par conséquent, les a regroupés.

En arabe, les mots de conjonction ne sont pas séparés du mot suivant. Généralement et merci "sert de lettre de signature en tant que" sincèrement " est utilisé en anglais. Le modèle appris que les deux mots {"et merci", "merci"} sont similaires, quelles que soient leurs formes.

Le deuxième exemple illustre une syntaxe spécifique caractéristique morphologique de l'arabe, où l'énumération des couples a sa propre forme. Les représentations de word embedding représentent un précieux ressource pour toutes les langues, mais en particulier pour langues à ressources limitées. On a démontré comment les mots imbriqués peuvent être utilisés dans le commerce, solution pour atteindre une performance élevée sur une tâche fondamentale de la TALN, et on croit que ces embarquements aideront les chercheurs à développer des outils dans des langues avec lesquelles ils n'ont aucune expertise.

6.3. Le travail de Ayah Zirikly et Mona Diab: Named Entity Recognition for Arabic Social Media [34]

La majorité des recherches sur l'arabe reconnaissance des entités nommé (R.E.N) s'adresse au la tâche pour le genre de fil de presse, où le la langue utilisée est l'arabe moderne standard (MSA), cependant, la nécessité d'étudier cette tâche dans les médias sociaux est de plus en plus vital. Les médias sociaux se caractérisent par l'utilisation à la fois de MSA et de l'arabe dialectal (DA), avec souvent un changement de code entre les deux variétés de langue. Malgré certaines caractéristiques communes entre MSA et DA, il existe d'importantes les différences entre lesquelles résultent en faible performances lorsque les systèmes de ciblage MSA sont appliqués pour R.E.N dans DA, aditionellement, la plupart des systèmes R.E.N reposent principalement sur répertoires géographiques, ce qui peut être plus difficile dans un contexte de traitement des médias sociaux en raison à une faible couverture inhérente. Dans ce travail, on a présenté un système R.E.N sans nomenclature pour les données dialectales qui donnent un score F1 de 72,68% ce qui est une amélioration absolue de 2 à 3% sur un état comparable, le système DA-NER basé sur une nomenclature d'art.

Jeux de données on a utilisé les ensembles de données de weblogs, microblogs et dialectal pour les expériences : ensemble de données Twitter, Jeu de données arabe dialectal (DA-EGY)

les données annotées ont été choisies parmi un ensemble de blogs Web qui sont identifiés manuellement par PMA comme dialecte égyptien et contient près de 40k jetons. Les données ont été annotées par un annotateur de langue maternelle arabe qui a suivi les directives de Consortium de Données Linguistiques pour le marquage.

On a aussi étudié l'impact des représentations de mots et intégration sur le système R.E.N arabe pour les données de médias sociaux.

6.4. Projet ARAVEC *(Abu Bakr Soliman, Kareem Eissa, Samhaa R. El-Beltagi)*[35]

AraVec est un projet open source de représentation de mots distribués pré-entraînés (incorporation de mots) qui vise à fournir à la communauté de recherche arabe en T.A.L.N, des modèles libres d'utilisation et puissants d'intégration de mots, ce projet a été réalisé par un groupe de chercheur dont Abu Bakr Soliman, Kareem Eissa, Samhaa R. El-Beltagy

- ***AraVec 1.0***

La première version d'AraVec fournit six modèles d'incorporation de mots différents construits sur trois domaines de contenu arabes différents ; Tweets, pages Web et articles

Wikipédia en arabe. Le nombre total de jetons utilisés pour construire les modèles s'élève à plus de 3 300 000 000. Ce document décrit les ressources utilisées pour la construction des modèles, les techniques de nettoyage des données utilisées, l'étape de prétraitement réalisée, ainsi que les détails des techniques de création d'intégration de mots utilisés.

AraVec est livré dans sa première version avec six modèles de Word Embedding différents construits sur trois domaines de contenu arabes différents ; Tweets Twitter Pages du World Wide Web Articles de Wikipédia en arabe : Au total, plus de 3 300 000 000 de jetons.

- ***AraVec 2.0***

La deuxième version d'AraVec fournit douze modèles d'intégration de mots différents construits sur trois domaines de contenu arabes différents ; Tweets, pages Web et articles Wikipédia en arabe. La différence entre cette version et la première réside dans le fait que l'hyper-paramètre de comptage minimal a été réduit à 50 au lieu de 500 pour le jeu de données Tweets, 200 pour le jeu de données de pages Web et 5 pour le jeu de données Wikipédia. Cela a abouti à des modèles qui couvrent davantage le vocabulaire. L'autre changement est la création d'un ensemble de six modèles d'incorporation dont la dimension est 100. Tweets Twitter Pages du World Wide Web Articles de Wikipédia en arabe Au total, plus de 3 300 000 000 de jetons.

- ***AraVec 3.0***

La troisième version d'AraVec fournit 16 modèles d'incorporation de mots différents construits sur deux domaines de contenu arabes différents ; Tweets et articles Wikipédia en arabe. La principale différence entre cette version et les précédentes est que nous avons produit deux types de modèles différents, les modèles unigrammes et n-grammes. Nous avons utilisé un ensemble de techniques statistiques pour générer les n-grammes les plus

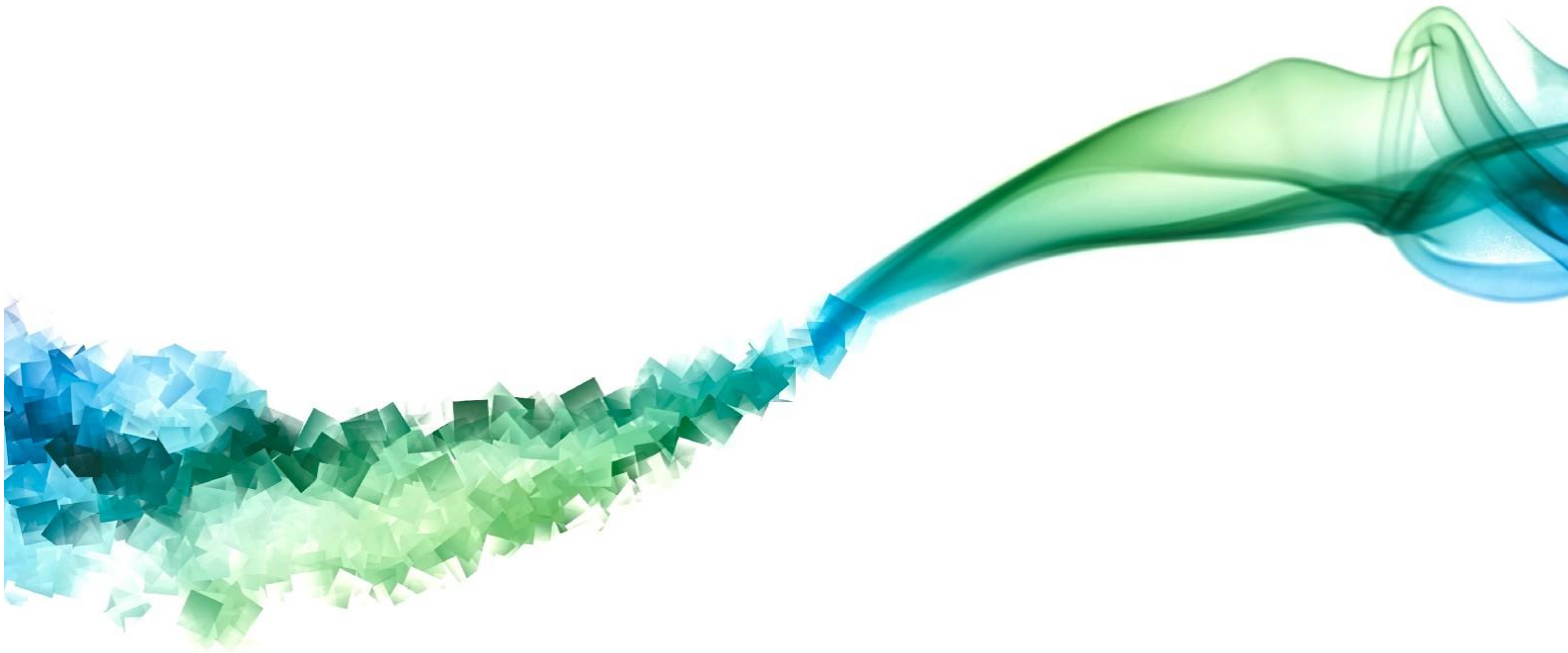
couramment utilisés de chaque domaine de données. Ces données sont été récupérés des sites internet (Tweets Twitter Articles de Wikipédia en arabe Par total des jetons de plus de 1 169 075 128 jetons.)

7. Conclusion

Les différentes recherches effectuées sur l'analyse de la morphologie de la langue arabe montrent que c'est une langue très difficile à traiter à cause de l'agglutination et des ambiguïtés graphiques.

Dans ce chapitre, nous avons explicité les différentes connaissances liées au représentations vectorielles des corpus arabe. On a commencé par une petite introduction puis on a cité les techniques relatifs à la représentation des mots (Word Embedding/Word2Vec), puis nous avons présenté la langue arabe et sa morphologie, ensuite, nous avons mis le point sur quelques travaux réalisés dans le contexte de représentation (Word Embedding) des corpus arabes.

Chapitre 03



Réalisation du model Word-embedding

1. Introduction

L'objectif principal de ce travail est de fournir des modèles de représentation Word embedding efficaces pour l'analyse des sentiments en arabe dans le domaine de l'hôtellerie. Dans cette partie nous allons d'abord expliquer notre stratégie pour la construction du model, (la récupération des données (avis) sous formes de fichiers textes regroupés et catégorisés dans un corpus puis le prétraitement des données afin de pouvoir les utiliser, puis la construction du model word2vec avec l'exposition de ses caractéristiques et les mesures de validation, d'évaluation et d'optimisation) ; puis nous allons exposer l'implémentation et l'évaluation dudit model conformément à la stratégie planifiée.

2. La stratégie du travail

On peut résumer notre stratégie de travail dans les points suivants :

*La récupération du corpus par scraping puis organisations des données dans des fichiers textes catégorisés.
*Le prétraitement des données : tokenization, normalisation et la construction du dictionnaire qui permet de ne pas prendre en compte des détails importants au niveau local (ponctuation, majuscules, conjugaison, …. etc.)
*La construction du model word2vec, le paramétrer, et l'évaluer avec des tests.

Fig.3.1 Le cycle de récupération et prétraitement d'un corpus de texte[1]

[1] source du schéma https://openclassrooms.com/fr/courses

2.1 Les outils utilisés :

2.1.1 Langage de programmation

Python :Est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions **[36]**.

Le langage Python est placé sous une licence libre proche de la licence BSD et fonctionne sur la plupart des plates-formes informatiques, des smartphones aux ordinateurs centraux, de Windows à Unix avec notamment GNU/Linux en passant par mac OS, ou encore Android, iOS, et peut aussi être traduit en Java ou .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser. La version utilisée dans notre travail : python 3.6.8

2.1.2 Les Bibliothèques [36] :

*** *Bibliothèque Beautiful soup* :** est un package Python permettant d'analyser des documents HTML et XML (y compris les balises malformées, c'est-à-dire les balises non fermées, nommées ainsi après balise soupe). Il crée un arbre d'analyse pour les pages analysées qui peut être utilisé pour extraire des données à partir de HTML, ce qui est utile pour le scraping Web.Il est disponible pour Python 2.7, Python 3.

***Bibliothèque NLTK* :** The Natural Language Toolkit NLTK, est une suite de bibliothèques et de programmes de traitement du langage naturel symbolique et statistique, écrit en langage de programmation Python. Il a été développé par

Steven Bird et Edward Loper, NLTK comprend des démonstrations graphiques et des exemples de données.

NLTK est destiné à soutenir la recherche et l'enseignement dans le domaine de la TALN, ou dans des domaines étroitement liés, notamment la linguistique empirique, les sciences cognitives, l'intelligence artificielle, la recherche d'informations et l'apprentissage automatique. Parmi ses fonctions (Tokenization, la manipulation des stop words, stop marks, Identifier les entités nommées …..) . NLTK a été utilisé avec succès comme outil d'enseignement, comme outil d'étude individuel et comme plate-forme de prototypage et de construction de systèmes de recherche.

***Bibliothèque Gensim* :**Gensim est une bibliothèque à code source libre pour la modélisation de sujets non supervisée et le traitement automatique du langage naturel, utilisant l'apprentissage statistique moderne développé par Radim Rehurek ,Gensim est implémenté en Python, et conçue pour gérer des collections de texte volumineuses à l'aide de la transmission en continu de données et d'algorithmes en ligne incrémentaux, ce qui le différencie de la plupart des autres progiciels d'apprentissage automatique qui ne ciblent que le traitement en mémoire.

Gensim permet l'analyse les documents textuels bruts ainsi que la recherche des structures sémantiques et la récupération des informations sémantiquement similaires, Gensim inclut des implémentations parallélisées en flux de fastText, algorithmes word2vec et doc2vec, ainsi que l'analyse sémantique latente (LSA, LSI, SVD), la factorisation matricielle non négative (NMF), l'attribution de Dirichlet latente (LDA), tf -idf et projections aléatoires **[37]**.
Cette bibliothèque est caractérisée par :
*La capacité de sa sémantique statistique évolutive
*L'analyse des documents en texte brut pour la structure sémantique
*La récupération des informations et concepts sémantiques similaires
En plus, la bibliothèque « Gensim » peut traiter de grands corpus à l'échelle du Web en utilisant des algorithmes d'apprentissage séquentiel récurrent.

2.1.3 L'environnement d'exécution

Jupyter : est une application web utilisée pour programmer dans plus de 40 langages de programmation, dont: Python, Julia, Ruby, R, ou encore Scala. Jupyter est une évolution du projet IPython. Jupyter permet de réaliser des calepins ou notebooks, c'est-à-dire des programmes contenant à la fois du texte en markdown et du code en Julia, Python, R... Ces notebooks sont utilisés en science des données pour explorer et analyser des données **[38]**.

***Google colab* :** Est un service cloud gratuit hébergé par Google afin d'encourager la recherche sur l'apprentissage automatique et l'intelligence artificielle, où l'obstacle à l'apprentissage et au succès réside souvent dans l'exigence d'une énorme puissance de calcul **[39]**.

Avantages de Google colab

- Le Colab est assez flexible dans sa configuration et fait la plupart des tâches lourdes à votre place.
- Prise en charge de Python 2.7 et Python 3.6
- Accélération GPU gratuite
- Bibliothèques préinstallées : toutes les principales bibliothèques Python telles que TensorFlow, Scikit-learn, Matplotlib, parmi beaucoup d'autres, sont préinstallées et prêtes à être importées.
- Construit sur le Jupyter Notebook
- Fonction de collaboration (fonctionne avec une équipe similaire à Google Documents): Google Colab permet aux développeurs d'utiliser et de partager le bloc-notes Jupyter entre eux sans avoir à télécharger, installer ou exécuter autre chose qu'un navigateur.
- Prend en charge les commandes bash
- Les blocs-notes Google Colab sont stockés sur le lecteur.

2.2 Le corpus

Le corpus est une collection de documents en langage naturel écrits ou parlés, stockés sur un ordinateur et utilisés pour déterminer comment la langue est utilisée. Plus précisément, un corpus est une collection informatisée systématique de langage authentique utilisée à la fois pour l'analyse linguistique et pour l'analyse de corpus **[40]**. Le corpus doit vérifier trois types de conditions **[41]** :

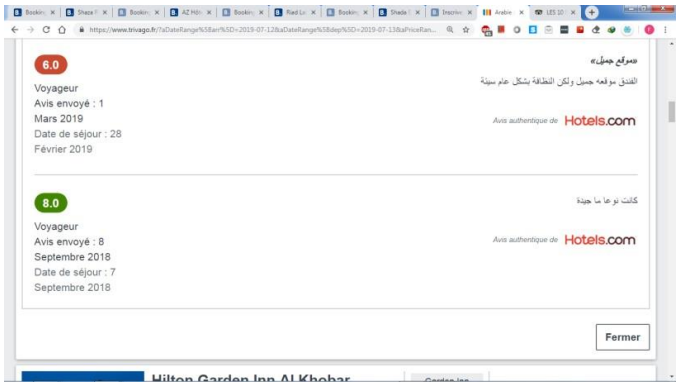
**Conditions de signifiance* : un corpus est constitué en vue d'une étude déterminée, portant sur un objet particulier,
**Conditions d'acceptabilité* : le corpus doit apporter une représentation fidèle, sans être parasité par des contraintes externes.

**Conditions d'exploitabilité* : les textes qui forment le corpus doivent être commensurables.

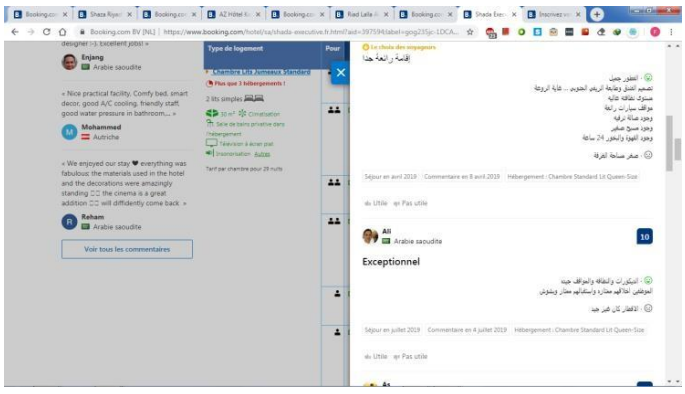
2.2.1. Sources des données

Le web est la source de la majorité des corpus réalisés, vu les opportunités qu'il offre (accessibilité aux données, Disponibilité et diversité des données, Adéquation des données en termes de quantité et de qualité, et la facilité relative de la récupération de ces données…)

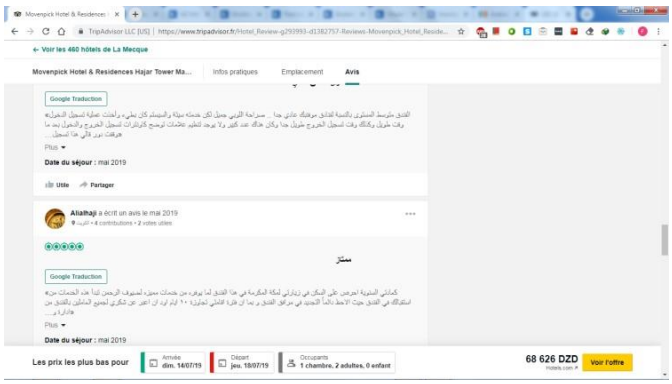
Notre domaine c'est l'hôtellerie, plus précisément l'hôtellerie arabe, donc notre cible c'est les sites de l'hôtelière ou les clients, internautes, visiteurs rédigent leurs commentaires, avis, évaluations, reviews, notes …. en arabe (MSA ou dialecte)



www.Trivago.com



www.Booking.com



www.Tripadvisor.com

Fig.3.2 Exemples des sites internet de l'hôtellerie les plus utilisés dans le monde arabe

2.2.2. Construction du corpus (le web scraping)

La première étape est la récupération du texte. Il existe plusieurs manières de récupérer du texte : soit depuis une base de donnée locale qu'on possède, soit depuis des fichiers XML ou autres, soit en scrapant des pages web comme le font les moteurs de recherches.

- ***Le web scraping***

Le Web scraping, la récupération Web est un raclage de données utilisé pour extraire des données de sites Web. Un outil de récupération Web peut accéder directement au Web via le protocole de transfert hypertexte ou via un navigateur Web. Bien que le Web scraping puisse être effectué manuellement par un utilisateur de logiciel, le terme désigne généralement les processus automatisés mis en œuvre à l'aide d'un robot ou d'un robot d'indexation Web. Il s'agit d'une forme de copie, dans laquelle des données spécifiques sont rassemblées et copiées à partir du Web, généralement dans un corpus, des fichiers textuels ou une base de données locale centralisée ou un tableur, pour une récupération ou une analyse ultérieure pour exploitation.

Certains sites Web utilisent des méthodes pour empêcher le Web scraping, telles que la détection et l'interdiction aux robots de parcourir (afficher) leurs pages. En réponse, il existe des systèmes de nettoyage Web qui utilisent des techniques d'analyse DOM, de vision par ordinateur et de traitement du langage naturel pour simuler la navigation humaine et permettre la collecte de contenu de page Web pour une analyse hors ligne.

- ***Les techniques du web scraping***

Les techniques actuels de Web scraping vont des solutions ad hoc, nécessitant des efforts humains, aux systèmes entièrement automatisés capables de convertir des sites Web entiers en informations structurées, avec des limitations, on peut citer entre autres (Copier-coller humain, Programmation http, Analyse HTML, Application et Logiciels…)

On peut également effectuer un le web scraping directement via un script écrit en langage de programmation (python, PHP, …) ce qui nous donne l'opportunité de personnaliser notre scrapping)

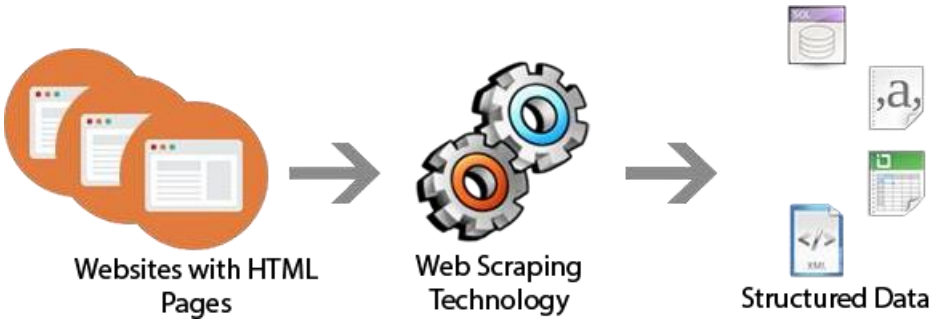


Fig.3.3 Le processus du web scraping[2]

A la fin de ce processus on obtient notre corpus, sous forme de fichiers catégorisé d'une façon précise afin de nous permettre une meilleure exploitation, dans notre cas on va repartir chaque avis suivant l'évaluation par étoile correspondante, de 01 à 05 onc on obtiendra un corpus constitué de 05 dossiers chacun comporte des avis de la même catégorie sous forme de fichier txt.

Le corpus est constitué de fichiers, chacun d'eux contient un nombre important des avis rédigés principalement en langue arabe

Le corpus, peut être organisé de plusieurs manière différentes :

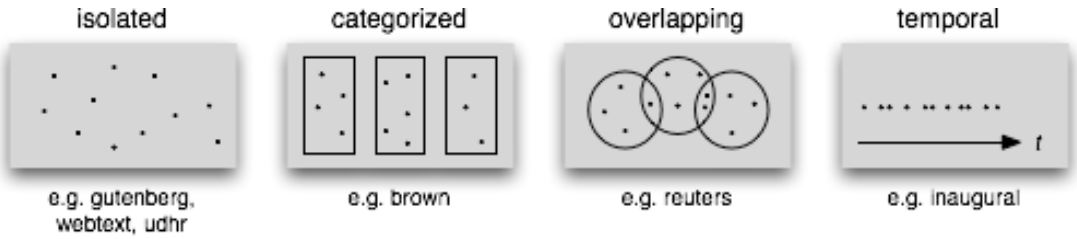


Fig.3.4 Les différents types de structuration du texte[3]

2.3 Prétraitement du corpus

Avant de soumettre le corpus au traitement pour la réalisation de notre model Word Embedding, il est nécessaire de passer par l'étape de prétraitement

Dans cette étape, nous effectuerons des opérations sur les données du corpus afin de les préparer pour la prochaine étape

2.3.1. Mise en forme

[2] Source du photo site what is web scraping : https://web-scraping/

[3] Source du photo site open class room : https://openclassrooms.com/fr/courses/4470541-analysez-vos-donnees-textuelles/4470548-recuperez-et-explorez-le-corpus-de-textes

Dans cette étape, on peut changer(convertir) le format de jeu de données avec lequel on va travailler TXT, JSON, CSV ….

2.3.2. Nettoyage

Dans cette étape, nous nettoyons les données. Par la suppression des données inutiles, aussi la suppression, ou la rectification des données ou des formes qui ne sont pas homogènes avec nos données par ex si on travaille sur la linguistique on peut supprimer les équations mathématiques car ils sont inutiles.

2.3.3. Échantillonnage

Dans cette étape, on va déterminer quels sont les attributs de données disponibles de notre corpus actuel et quels attributs de données peuvent être dérivés par nous (mots, phrases…)

Les étapes précédentes sont des étapes de base pour préparer l'ensemble de données au prétraitement linguistique.

2.3.4. Prétraitement linguistique

On entend par « *Prétraitements du Texte* **»** (PTT) ou *« Text Processing Stage* » (TPS) : l'étape qui vient juste après la récupération des données, elle consiste en la *préparation* du texte brute (comme il le présente l'émetteur d'opinion) pour qu'il soit exploitable par la prochaine phase (représentation word2vec).

La PTT peut être effectués avec des techniques d'analyse linguistique plus ou moins poussées et des coûts en temps et en ressources très variables. Elle consiste à normaliser les diverses manières d'écrire un même mot, à corriger les fautes d'orthographe évidentes ou les incohérences et expliciter les ambiguïtés.

-Le format variable d'encodage des textes, par exemple des caractères encodés de manières différentes d'un texte à l'autre, constitue un premier type d'incohérence. Ce type d'incohérence est à traiter en premier lieu car le problème se situe au niveau du caractère et il faut décider d'un encodage unique avant de pouvoir effectuer des traitements au niveau des mots.

-La présence des fautes d'orthographe constitue un autre type d'incohérence. Ce traitement devrait être effectué avant celui des ambiguïtés lexicales pour limiter les erreurs dues aux données erronées.

-Les ambiguïtés lexicales nécessitant parfois des ressources externes aux textes afin de les lever, par exemple un lexique d'abréviations, constitue un premier type d'ambiguïté. Ce traitement devrait être effectué avant celui de la structure des textes pour ne plus avoir le problème des abréviations.

Donc le PTT est primordial mais elle se diffère entre les modèles parce qu'ils doivent mettre le texte sur un format bien définit qui dépend évidemment de l'objectif final. Cependant on a pu noter qu'il y a une grande partie de taches de la PTT qui sont communes entre toutes les approches qu'on a examinées, cette partie se résume dans les points suivants :

-Séparation des entités lexicale du texte (Tokenisation) en éliminant les stop-mots (stop- words) : les blancs " ", les séparateurs (, ; ? . etc.) et même des prépositions comme : {à, dans, au, ou, on …etc.} : (français), {at, in, the, on …etc.} (anglais), {في، عن، من} (arabe). Enfin, on récupère chaque entité à part dans une structure de donné au choix, ces entités sont généralement connues par le nom : token(T).

-Elimination des lettres répétées, par exemple le token : bon couraaaage, (a) beaucouuuuuups (u) hellooooo (o) (salut) "مرحبا١١١١"(ا) (bienvenu) ces mots deviennent successivement juste : bon courage, beaucoup , hello, et "مرحبا",

-Normalisation, qui est la tâche la plus importante dans le PTT, surtout pour une langue MRL (*Morphologically Rich Languages)* comme l'Arabe **[42]**.

-Racinisation ou Désuffixation ou Lemmatisation (Stemming) qui consiste en un procédé de transformation des mots en leur radical ou racine (stem).

Les deux dernières taches (la normalisation et la racinisation) demandent plus d'explication, ce qui nécessitera l'ajout des deux paragraphes suivants :

Normalisation : La normalisation est une tâche qui a pour but de proposer, pour chaque mot considéré comme « fautif », sa forme normativement correcte ou une forme qui lui est flexionnellement liée. Nous considérons qu'une forme est « fautive » si son orthographe peut gêner une analyse ultérieure. La normalisation du texte avant son stockage ou son traitement permet de séparer les problèmes, car la cohérence de la saisie est garantie avant toute opération. La normalisation de texte nécessite de savoir quel type de texte doit être normalisé et comment il doit être traité par la suite. Il n'y a pas de procédure de normalisation universelle.

*Exemple : les mots engagemt, changemt (forme fautive) –engagement, changement (forme correcte)

Dans la langue Arabe une lettres peut avoir plusieurs formes différentes par rapport à sa position dans le mot, en outre quelques lettres dans la langue Arabe peuvent prendre plusieurs formes non pas seulement par rapport à leurs position mais le contexte peut modifier la forme de la lettre, par exemple la lettre "أ" (Alif) a quatre formes différentes "أ", "إ", "ا", "آ", mais malheureusement même les locuteurs natifs de l'Arabe confondent ces formes et parfois il arrive ou le

sens du mot change carrément, comme pour "أكل" (Il a mangé) et "آكل" (quelque un qui mange), donc le premier mot est un verbe alors que le deuxième est un nom. Le premier est un substitut grammatical. Dans ce cas la majorité des approches propose de simplifier la représentation des différentes formes pour une lettre en une seule représentation, par exemple toutes les formes du "أ" (Alif) se réduit en "ا" avant de passer à la classification.

Racinisation (Lemmatisation) : La lemmatisation est par définition une action consistant à l'analyse lexicale d'un texte avec pour but de regrouper les mots d'une même famille. On parle ici de donner la forme canonique d'un mot ou d'un ensemble de mots : Chacun de ces mots d'un contenu donné se trouve réduit en une entité appelée en lexicologie lemme ou encore « forme canonique d'un mot ». Les lemmes d'une langue utilisent plusieurs formes en fonction du : Genre (masculin ou féminin), Nombre (un ou plusieurs), Personne (moi, toi, eux…) et Mode (indicatif, impératif…)

Le lexique Arabe comprend trois catégories de mots : verbes, noms et particules. Les verbes et les noms sont le plus souvent dérivés d'une racine à trois lettres radicales ; Une famille de mots peut ainsi être générée d'un même concept sémantique à partir d'une seule racine à l'aide de différents schèmes (structure ou forme) **[43]**. Ce phénomène est une caractéristique à la morphologie Arabe. On dit donc que l'Arabe est une langue à racines réelles à partir desquelles, on déduit le lexique Arabe selon des schèmes qui sont des adjonctions et des manipulations de la racine **[44]**.

Le tableau suivant, donne quelques exemples de schèmes appliqués au mot « كتب » (écrire). On peut ainsi dériver un grand nombre de noms, de formes et de temps verbaux.

Schèmes	**كتب**	**Notion d'écrire**
فاعل	**كاتب**	**Ecrivain**
فعل	**كـتـب**	**A écrit**
مفعل	**مكتب**	**Bureau**

Tableau 3.1 Exemple de schèmes pour le mot كتب (écrire)

Pour détecter la racine d'un mot, il faut connaître le schème (الوزن) par lequel il a été dérivé et supprimer les éléments flexionnels (antéfixes, préfixes, suffixes et post fixes) qui ont été ajoutés, parce qu'en Arabe un mot peut signifier toute une phrase grâce à sa structure composée qui est une agglutination d'éléments de la grammaire. Cependant, Il existe deux méthodes morphologiques qui sont proposés dans la littérature pour la racinisation pour l'Arabe : la méthode basé-racine *root-based-methode* (par exemple, Khoja stemmer) et la méthode *Light-*

stemming. La première méthode supprime tous les affixes et retourne chaque mot arabe à son modèle de racine **[45]**. Alors que la seconde méthode élimine seulement les affixes (préfixes communs et suffixes) sans modifier l'origine (*root*) d'un mot **[46]**.

En arabe on a trouvé que la majorité des approches recommande la seconde méthode. La principale raison de ce choix est que beaucoup de mots qui partagent la même racine peuvent avoir des significations complètement différentes ainsi que des sentiments opposés. Par exemple, les deux mots suivants "اللاعبون" et "يتلاعب" qui signifie respectivement « les joueurs » et« manipule » ont la même racine "لعب" qui signifie «jouer » mais sémantiquement ils sont très différents. En revanche, si nous appliquons le premier algorithme les affixes seront tout simplement retirés et le résultat sera donc "لعب" qui signifie «jouer» ce qui induit en erreur lors de la classification, mais si nous appliquons la méthode *Light-stemming* on n'aura pas ce problème.

2.4 La réalisation du model Word embedding :

Notre corpus est prêt pour l'étape finale : celle de la réalisation du model word2vec

2.4.1 Techniques

Les modèles d'espace vectoriel (VSM) sont l'un des schémas de représentation de texte les plus anciens et les plus connus. Traditionnellement, le modèle spatial vectoriel a été principalement utilisé pour la représentation de documents, avec des travaux plus récents on a pu étendre ce modèle à la représentation de mots ou de termes. Dans ces travaux récents, les mots sont représentés de manière continue, espace où les mots sémantiquement similaires ont une mesure de similarité élevée dans cet espace. Les VSM s'appuient sur « Hypothèse de distribution », qui indique que les mots utilisés dans les mêmes contextes ont généralement une signification similaire.

Les deux approches principales pour construire ces représentations sont : les approches basées sur le nombre (approches Countbased) et les approches prédictives.

* Approches Countbased : calculent les statistiques de cooccurrence entre les mots puis mappent ces statistiques en un vecteur dense chaque mot.
* Approches prédictives : tentent de prédire un mot de ses voisins en termes de vecteur dense appris pour chaque mot.

Le terme « Word embedding » a été inventé pour la première fois par Bengio et al **[47]**. Le modèle proposé reposait sur l'idée d'obtenir les valeurs pour les

vecteurs de mots ou les intégrations en formant un modèle de langage neuronal. En 2008, Collobert et Weston **[48]** ont montré que l'intégration de mots était un outil efficace dans de nombreuses tâches en aval. C'était Mikolov et al **[49]**. qui ont propulsé l'idée à la pointe de la recherche et ont contribué à sa généralisation par le biais de la création de la boîte à outils Word2Vec qui peut être facilement utilisée et ajustée pour générer des incorporations. Mikolov et al. ont proposé deux architectures de modèle différentes pour représenter les mots dans un espace vectoriel multidimensionnel, à savoir le modèle de (CBOW) et le modèle de skip-gram.

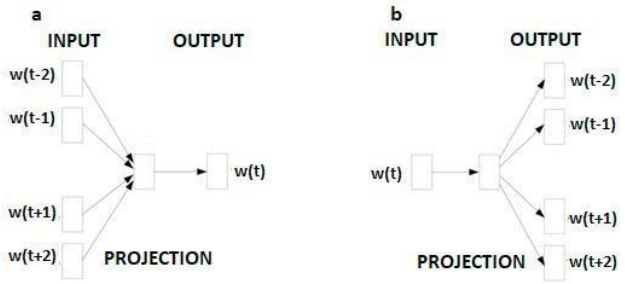


Fig.3.5 Les algorithmes Word2vec (a) CBOW, (b) Skip-gram

2.4.2. Construction du modèle

Les modèles que nous avons réalisés ont été construits en utilisant l'outil Gensim, qui est une boîte à outils efficace créée pour traiter de nombreuses tâches NLP courantes et qui inclut une implémentation pour le modèle Word2Vec.

Notre travail propose 02 modèles d'incorporation de mots différents, dans lesquels chaque domaine de texte (avis) a deux modèles différents ; l'un construit en utilisant la technique CBOW et l'autre en utilisant la technique Skip-Gram. Pour Construire ces modèles, nous avons effectué un grand nombre d'expériences pour déterminer les valeurs optimales des hyper paramètres :

**La taille de la fenêtre [window_size] :* les mots de contexte sont des mots voisins du mot cible, mais jusqu'où ou près ces mots devraient-ils être pour être considérés comme voisins ? si on prend la valeur 2, ça signifie que les mots qui sont 2 à gauche et à droite des mots cibles sont considérés comme des mots de contexte.

**la taille du vecteur [Size n]* : C'est la dimension du mot incorporant et elle varie généralement de 100 à 300 en fonction de la taille de votre vocabulaire. Les dimensions au-delà de 300 ont tendance à avoir un avantage décroissant

le seuil minimal [Min_count] : Ignorer tous les mots dont la fréquence totale est inférieure à celle-ci.

Le nombre d'itérations[iter] : C'est le nombre d'itérations au cours d'entrainement. À chaque itération, nous parcourons tous les échantillons d'entraînement.

Les phrases iterables [Sentence] : peuvent être simplement une liste de listes de token, mais pour les corpus plus grands.

Chemin d'accès à un fichier de corpus au format [LineSentence] : *corpus_file*, on peut utiliser cet argument au lieu de sentence pour améliorer les performances. Un seul des phrases ou des arguments corpus_file doit être passé (ou aucun d'eux, dans ce cas, le modèle n'est pas initialisé).

Nombre de thread [Workers] : pour utiliser ces nombreux threads de travail afin de former le modèle (= formation plus rapide avec des machines multi cœurs).

Algorithme utilisé *({0, 1}) :* 1 pour skip-gram ; sinon CBOW

3. Implémentation

3.1 La récupération des données

Le site internet cible du scraping c est « https://www.tripadvisor.fr/ »

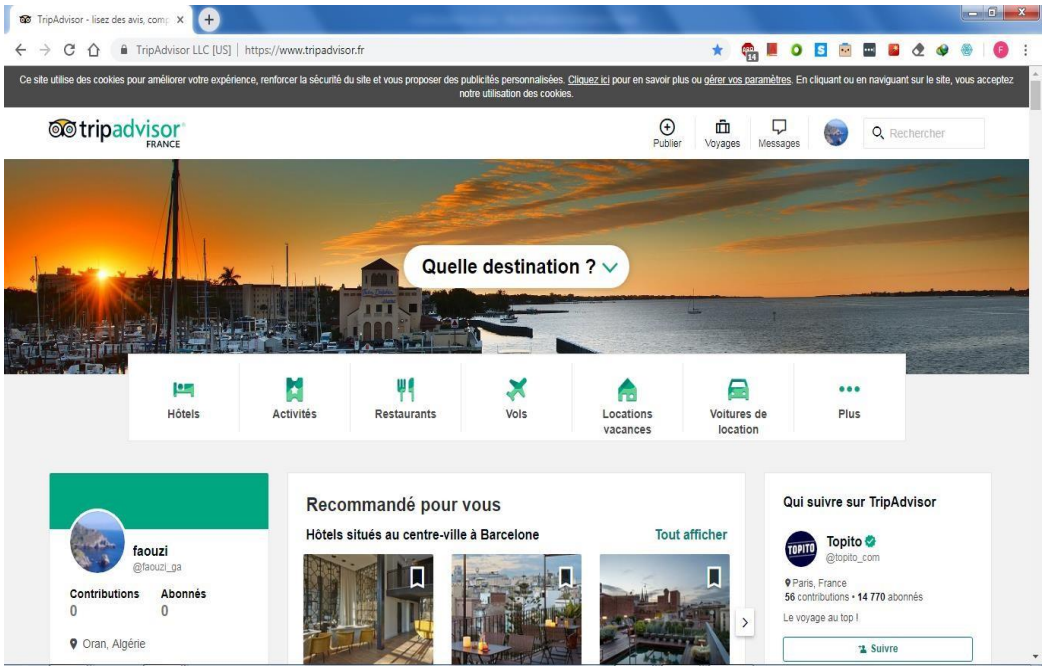


Fig.3.6 Interface du site TripAdvisor

La collectes des avis rédigés en arabe se fait par un script python en utilisant la bibliothèque Beautiful soup, récupérant ces avis on les catégorise par leurs évaluations conformément au nombres des étoiles correspondants.

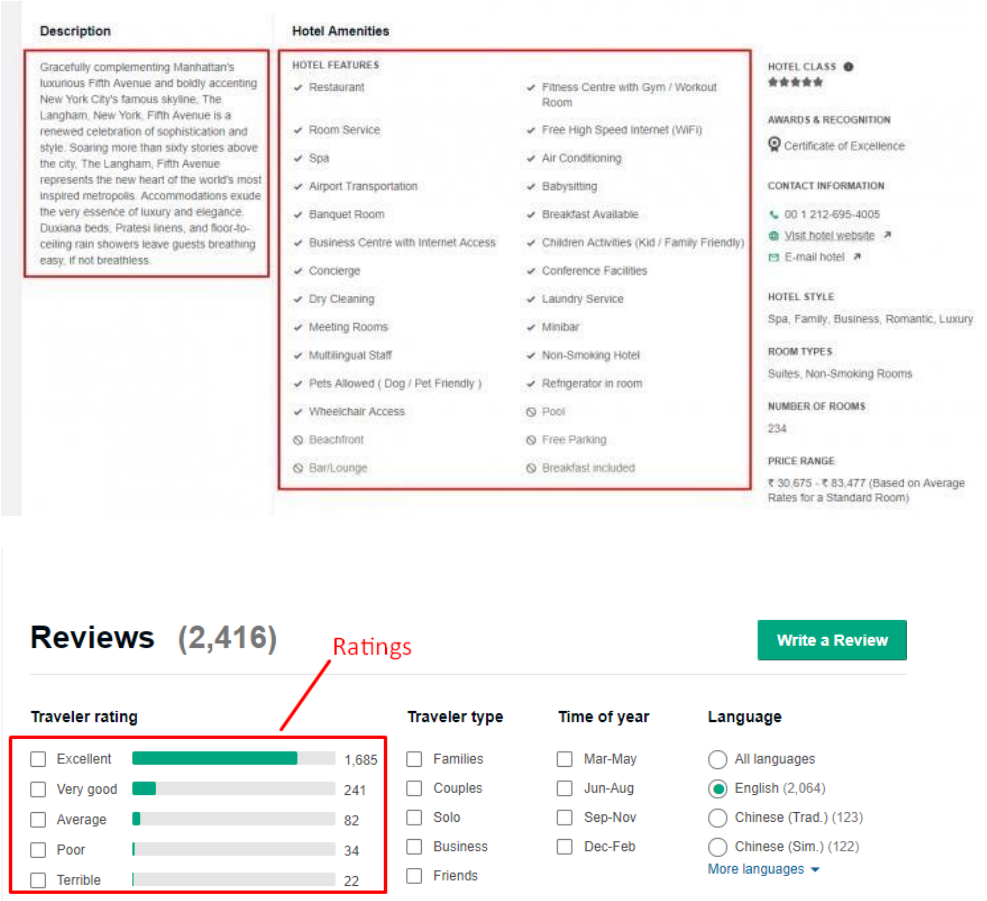


Fig.3.7 Structure du site TripAdvisor

3.1.1 Le scraping

** Installation et Importation des bibliothèques*

Identifier la structure du sites HTML

Installer Beautiful Soup et Requests (*pip install beautifulsoup4 pip install requests)*

import requests , from bs4 import BeautifulSoup , import csv

import webbrowser , import io

**Scraping le site*

```
page_link = 'https://www.tripadvisor.fr/'
page_response = requests.get(page_link, timeout=5)
page_content = BeautifulSoup(page_response.content, "html.parser")
textContent = []
for i in range(0, 20):
```

** Isolation des résultats*

```
paragraphs = page_content.find_all("p")[i].text textContent.append(paragraphs)
```

Les résultats ressemblent davantage à ceci :

استثنائي .كل الامور ممتازة لاسيما الطعام .وخدمة الغرف السريعة .والهدوء .سكنا خمسة ايام ولم نستفد من خدمة المسبح لطول مدة الصيانة ". الأصالة و الإبداع ."تصميم الفندق رائع ، إدارة ممتازة ، طاقم العمل ممتعا ". فندق ممتاز ."قربه من المسجد وتعاون الموظفين عدم وجود الكحوليات المسبح مكيف داخل المبنى .صغر الموقف أمام البوابة

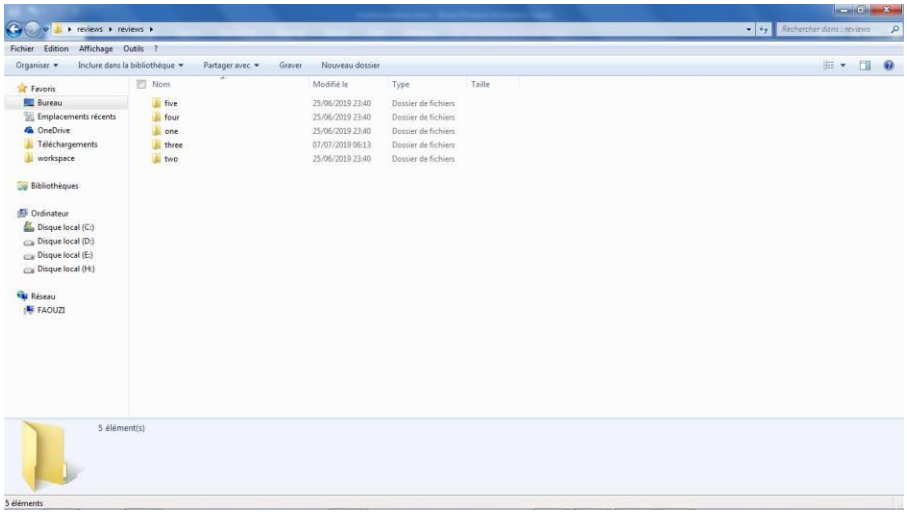


Fig.3.8 Structure du corpus (avis récupérés et repartis en 05 dossiers)

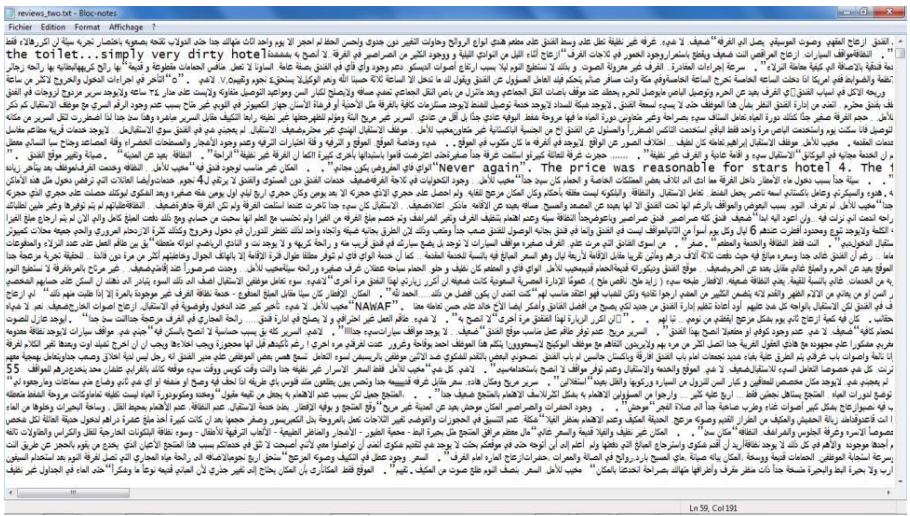


Fig.3.9 Le contenu de chaque fichier txt

On a récupéré les avis rédigés en arabe des utilisateurs /internautes du site de l'hôtellerie TripAdvisor, et on les a classés par catégories, sous forme de fichiers txt. Ces avis sont à l'état brut et nécessitent un prétraitement avant de les exploiter.

Le scraping du contenu arabe est plus difficile vu la spécificité de cette langue, sa morphologie et son encodage.

3.2 Prétraitement

Après récupérer les donnes sous forme de textes, ces données vont subir un prétraitement conformément à la stratégie.

On peut citer quelques opérations de prétraitement dans ce qui suit :

Suppression des stop words

```
arab_stop=set(nltk.corpus.stopwords.words("arabic"))
   r=sent_tokenize(File.read())

W = " كان علي ان اتجه الى الفندق ..... الأسعار الان جد مغرية في تونس..... "
print W : اتجه الفندق .....الاسعار مغرية تونس
```

Suppression des diacritiques

```
def remove_diacritics(string):
regex = re.compile(r'[\u064B\u064C\u064D\u064E\u064F\u0650\u0651\u0652]')
  return re.sub(regex, '', string)

W =  " مَشْفىٰ "
W = remove_diacritics(W)
print (w)  :  مشفى
```

Suppression des lettres repetés:

```
import re
   re.sub(r'(\w){5}', r'\1', 'راااائع')   :  رائع
```

Suppression des URLs

```
def remove_urls(string):
  regex = re.compile(r"(http|https|ftp)://(?:[a-zA-Z]|[0-9]|[$ _@.&+]|[!*\(),]| (?:%
[0-9a-fA-F][0-9a-fA-F]))+")
  return re.sub(regex, ' ', string)

W =" https://www.tripadvisor.fr/Attraction_Review-g293734-d477277-Reviews-
Jardin_Majorelle-Marrakech_Marrakech_Tensift_El_Haouz_Region.html حديقة
تمتاز بكثافة الاشجار وغرابتها .. مرتبة وانيقة استحقت الزيارة ومقهى لذيذ.. والمتحف الامازيغي داخلها
جدير بالزيارة وزيادة الثقافة عن الامازيق وحضارتهم.
الجديد "
W = remove_urls(W)
Print(W) :  حديقة تمتاز بكثافة الاشجار وغرابتها .. مرتبة وانيقة استحقت الزيارة ومقهى لذيذ..
والمتحف الامازيغي داخلها جدير بالزيارة وزيادة الثقافة عن الامازيق وحضارتهم.
```

Suppression des nombres

```
def remove_numbers(string):
   regex =
re.compile(r"(\d|[\u0660\u0661\u0662\u0663\u0664\u0665\u0666\u0667\u0668\
u0669])+")
   return re.sub(regex, ' ', string)
```

```
W ="الفندق روعة وتصميمه جميل ... الغرف بمساحة اكثر من 20 م .. واطلالته ع المسبح .. يتبع
   "الفندق مول وسوبر ماركت يبعد 200 م.. والاسواق قريبه 800 م .. النت متوفر وسريع 2 جيجا ..
W = remove_numbers(W)
Print (W) : الفندق روعة وتصميمه جميل ... الغرف بمساحة اكثر من  م .. واطلالته ع المسبح .. يتبع
جيجا ..  الفندق مول وسوبر ماركت يبعد م.. والاسواق قريبه م .. النت متوفر وسريع
```

**** Normalisation d'une chaîne***

```
def noramlize(string):
   regex = re.compile(r'[إأآا]')
   string = re.sub(regex, 'ا', string)
   regex = re.compile(r'[ى]')
   string = re.sub(regex, 'ي', string)
   regex = re.compile(r'[ؤئ]')
   string = re.sub(regex, 'ء', string)
   return string
```

il s'agit d'unifier les différents formes un lettre
les lettres (إ أ آ ا) remplacés par le lettre (ا)
les lettres (ؤ ئ) remplacés par le lettre (ء)
les lettres (ى ي) remplacés par le lettre (ي)

```
W = " أفضل الإقامة..... الآتية ..... لأني ..... المؤداة..... بلد المنشئ .....الملتقى
W = noramlize(W)
Print(W)  : افضل الاقامة..... الاتية ..... لاني ..... المءداة..... بلد المنشء .....الملتقي
```

**** Suppression des mots non arabes***

```
def remove_non_arabic_words(string):
   return ' '.join([word for word in string.split() if not re.findall(
r'[^\s\u0621\u0622\u0623\u0624\u0625\u0626\u0627\u0628\u0629\u062A\u062
B\u062C\u062D\u062E\u062F\u0630\u0631\u0632\u0633\u0634\u0635\u0636\u
0637\u0638\u0639\u063A\u0640\u0641\u0642\u0643\u0644\u0645\u0646\u064
7\u0648\u0649\u064A]', word)])
```

```
W = "  الفندق رائع جدا  so cool  …  خدمة الانترنت  wifi …….  الاطلالة جميلة top "
W = remove_non_arabic_words(W)
Print(W)  :  الفندق رائع جدا ..... خدمة الانترنت ..... الاطلالة جميلة
```

Suppression des espaces supplémentaires

```
def remove_extra_whitespace(string):
   string = re.sub(r'\s+', ' ', string)
   return re.sub(r"\s{2,}", " ", string).strip()
```

```
W = " فندق اكثر من  رائع ، الغرف  و الضيافة و  مكان  الفندق  مميز  جدا "
W = remove_extra_whitespace(W)
Print(W)  :  فندق اكثر من رائع، الغرف و الضيافة و مكان الفندق ممز جدا
```

**** Suppression les symboles non arabes***

```
def remove_non_arabic_symbols(string):
   return re.sub(r'[^\u0600-\u06FF]', ' ', string)
```

```
W = " قربه من دبي مول ومن محطه المترو☺ ☺ ☺ ☺  كل شيء في هذا الفندق ممتاز .
" 5 دقائق كنت في الغرفة...نظافة المكان واتساعه...@ الاستقبال مع ابتسامه عريضة .👈 👈 👈
W = remove_non_arabic_symbols(W)
Print(W) : الاستقبال مع ابتسامه .كل شيء في هذا الفندق ممتاز. قربه من دبي مول ومن محطه المترو
عريضة 5 دقائق كنت في الغرفة...نظافة المكان واتساعه...
```

3.3 Construction du model Word Embedding

3.3.1 Les hyper paramètres

Après la réalisation du model on a pu déterminer les valeurs optimales des hyper paramètres de notre modèle qui permettent d'obtenir les meilleurs résultats

*La taille de la fenêtre *[window_size] :* 3
*la taille du vecteur *[Size n]* 300
* le seuil minimal *Min_count* : Ignorer tous les mots dont la fréquence totale est inférieure à celle-ci.
*Le nombre d'itérations *iter* :20.
*Nombre de thread *Workers :8.*
*Algorithme utilisé 0 *CBOW*

3.3.2 Code

```
print(sentence[0:10])
model=word2vec(sentence,min_count=1,size=300,iter=20,sg=0,workers=4)
model.wv.save_word2vec_format('model.txt')
print("fin")
```

Le modèle prenait 01 heures pour s'entraîner sur un PC Intel i7-3770 à 3,4 GHz Quad core, avec 08 Go de RAM, sous Windows 7.

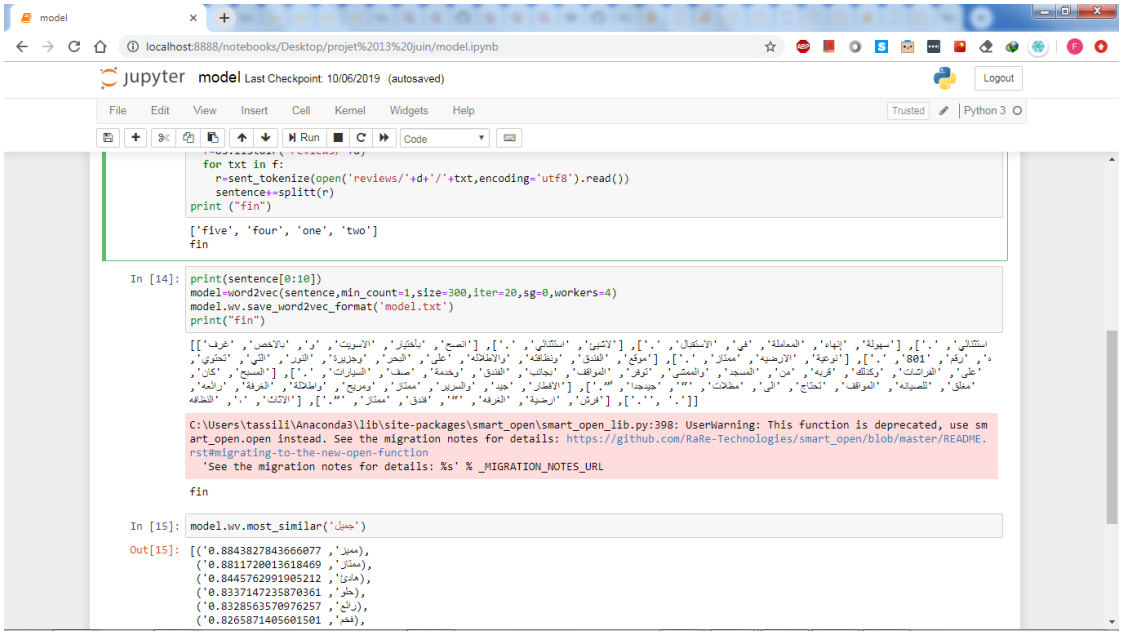


Fig.3.10 Code de construction du model sur Google Colab

*Enregistrement du modèle : model.save('trained_word2vec.bin')
vocab = model.wv.vocab

*Le nombre de mots constituant le vocabulaire model.corpus_count

Le nombre de mots constituant le vocabulaire de notre corpus (1873 mots)

* Echantillon du vocabulaire généré

list(vocab)[:10]

[[' استثنائي', '.'], 'بأختيار', 'انصح', '.'], 'استثنائي', 'لاشيئ', '.'], 'الاستقبال', 'في', 'المعاملة', 'إنهاء', 'سهولة', '.'], 'الاسويت', 'و', 'بالاخص', 'غرفه', 'رقم', ' 801 ', '.'], 'ممتاز', 'الارضيه', 'نوعية', '.'], 'موقع', 'الفندق', 'ونظافته',
'والاطلاله', 'على', 'البحر', 'وجزيرة', 'النور', 'التي', 'تحتوي', 'على', 'الفراشات', 'وكذلك', 'قربه', 'من', 'المسجد', 'مغلق', 'كان', 'المسبح', '.'], 'السيارات', 'صف', 'وخدمة', 'الفندق', 'بجانب', 'المواقف', 'توفر', 'والممشى', 'للصيانه',
'الموافق', 'تحتاج', 'الى', 'مظلات', '"', 'جيدجدا', 'الافطار', '.'], 'جيد', 'والسرير', 'ممتاز', 'ومريح', 'واطلالة', 'الغرفة', 'رائعه', '.'], 'الاثاث', '،', 'النظافه', 'الغرفة', 'رائعه', '.'], 'فرش', 'ارضية', 'الغرفه', '"', 'فندق', 'ممتاز', '"', '.'], '.']]

4. Evaluation du model :

L'étude de l'efficacité du model dépend des paramètres dont le changement peut entraîner des modifications importantes **[50]**. Ces paramètres comportent la méthode de construction de modèles (CBOW ou SKP-GRAM), les hyper paramètres de ces méthodes, le corpus d'apprentissage (nature, taille, prétraitements) **[51]** et les types de contextes utilisés pour représenter les unités lexicales.

L’évaluation du modèle généré, peut être fait par deux méthodes qualitative et quantitative **[52].**

4.1 Évaluation qualitative

L’évaluation qualitative est basée sur les mesures de similarité **[53]**, ces mesures peuvent être présentés sous différents formes soit la similarité entre deux mots ou entre deux groupes (clusters). On a testé notre model sur un très petit sous-ensemble de (mots/groupe de mots/ d'entités nommées) et on a appliqué un algorithme de regroupement pour voir si les mots proches contextuellement se regroupent ou non.

4.1.1 Similarité

- **Les mots similaires à un mot donné**

Extraction des mots les plus proches sémantiquement d’un mot donné

model.wv.most_similar('جميل ')

[('مميز ', 0.8843827843666077), ('ممتاز ', 0.8811720013618469), ('هادئ ', 0.8445762991905212), ('حلو ', 0.8337147235870361), ('رائع ', 0.8328563570976257), ('فخم ', 0.8265871405601501), ('هادي ', 0.8072738647460938), ('متميز ', 0.7943137884140015), ('روعه ', 0.7938087582588196), ('واسع ', 0.7929916381835938)]

On a utilisé le model pour établir une liste de mots similaires (05 mots) à un mot choisi comme le montre le tableau 3.2

Mot choisis	Mot similaire prédits
جميل	مميز ، ممتاز ، حلو ، رائع ،فخم

Tableau 3.2 Mots similaires à un mot donné

Par rapport le model Aravec 3.0 notre modèle est un peu spécial vu la nature de son vocabulaire (Hôtellerie), ce qui va être reflété sur les résultats des mots similaires, le tableau 3.3 comporte les 05 mots similaires à un mots choisis, la liste de ces 05 mots a été établit à l’aide du model Aravec 3.0 puis à l’aide de notre model

Mot	طعام
Model Aravec 3.0	، الاكل،الطبخ،الشراب، الصحة،البدانة
Notre model	بوفيه ،مطعم ،الإفطار، الاكل ،المنيو

Tableau 3.3 Mots similaires établit par le model Aravec et notre model

Les différences notées sont dues au spécificité du corpus (vocabulaire), et peut être considérés comme avantage si le model est utilisé dans des taches relatives au TALN notamment l’analyse des sentiments

- **Vérification du taux de similarité entre deux mots**

Vérifier à quel point deux mots sont proches sémantiquement ; on choisit deux mots qui appartiennent à notre vocabulaire, et on applique notre model.

```
similarity = word_vectors.similarity ('سيارة ', 'موقف ')
similarity > 0.8
True
```

- **Vérification du taux de similarité entre deux groupes de mots**

Opération similaire a la précédente mais avec deux groupes de mots au lieu de deux mots

```
sim = word_vectors.n_similarity(['مطعم ', 'فندق '], ['بوفيه ', 'غرفة '])
print("{:.4f}".format(sim))
0.7067
```

4.1.2. Regroupement (clustering) des mots

- **Regroupement sémantiques des mots**

Le regroupement des mots (clustering) basée sur la sémantique, une liste des mots qui appartiennent à notre vocabulaire peut être regroupés suivant leurs sémantiques.

Groupe 1	غرفة ، جناح ،،طابق،
Groupe2	بوفيه،طعام،إفطار،غداء، المنيو

Tableau 3.4 Regroupement sémantique des mots

Pour le regroupement des mots on a utilisé l'algorithme de clustering K-Means (avec k = 2). Pour la représentation des clusters on a utilisé la méthode stochastique t-distribuée avec outil d'incorporation de voisins (t-SNE) d'Ulyanov, permettant de visualiser et d'examiner les résultats dans un graphique en 2D.

Fig.3.11. Regroupement (clustering) sémantique des mots

- **Regroupement des mots suivant leur polarité**

Une liste des mots dont la polarité est plus ou moins intense a été choisi pour classification suivant l'orientation sémantiques. (Les mots appartiennent à notre vocabulaire) les résultats figurent sur le tableau 3.5 où les mots de même polarité ont été regroupés.

Positive	رائع ، فخم ،حلو ،ممتاز
Negative	متواضع ،ضعيف ،سيء،رديء

Tableau 3.5 Regroupement des mots suivant leur polarité

Pour le regroupement des mots on a utilisé l'algorithme de clustering K-Means (avec k = 2). Pour la représentation des clusters on a utilisé la méthode stochastique t-distribuée avec outil d'incorporation de voisins (t-SNE) d'Ulyanov, permettant de visualiser et d'examiner les résultats dans un graphique en 2D

Fig.3.12 Regroupement (clustering) des mots suivant leur polarité

On peut constater, à travers l'exemple que, les mots de même polarité ont été regroupés.

4.1.3 Regroupement d'entités nommées

Tel que l'opération de classification suivant la polarité, la classification des entités nommées choisis, suivant leurs catégories (que nous avons limité au 03 fonction, location, saison)

Classe	Mots
Fonction	عامل ،موظف،المدير ،مسؤول
Location	دبي ،الغردقة ،مراكش، القاهرة
Saison	الصيف،موسم،الربيع ،التنزيلات

Tableau 3.6 Regroupement des entités nommées

Pour cette fonctionnalité notre modèle est limité par son vocabulaire on a constaté qu'il présente des limites notamment dans les cas où les classes ou les mots ne sont pas relatifs à notre domaine d'étude, par ailleurs le model Aravec3.0 s'est montré plus efficace à cause de son vocabulaire plus global

Pour le regroupement des mots on a utilisé l'algorithme de clustering K-Means (avec k = 3). Pour la représentation des clusters on a utilisé la méthode stochastique t-distribuée avec outil d'incorporation de voisins (t-SNE) d'Ulyanov, permettant de visualiser et d'examiner les résultats dans un graphique en 2D

En examinant les résultats, nous pouvons constater que les modèles prennent en compte les similitudes entre les entités nommées

Fig.3.13 Regroupement (clustering) des entités nommées

4.1.4 Prévision d'un mot suivant le contexte

On peut prévenir le mot qui manque à une liste de 03 mots données, en utilisant une analogie entre ces mots

result = word_vectors.most_similar(positive=[' فطور ', ' صباح '], negative=[مساء])
print("{}: {:.4f}".format(*result[0]))
عشاء : 0.7699

Le tableau 3.7 montre le résultat.

Mots données	(صباح ، فطور) ، مساء
Prévisions	عشاء

Tableau 3.7 Prévisions du mot manquant

4.1.5 Détection de mot intrus

Dans un groupe de mots donné on peut prédire le mot qui n'appartient pas à ce groupe (suivant sa sémantique) comme le montre le résultat présenté dans le tableau 3.8

Groupe de mots	غرفة ، طابق ، جناح ،
Mot intrus	سريع

Tableau 3.8 Détection du mot intrus

4.2 Évaluation quantitative

L'évaluation quantitative est basée sur une mesure de comparaison de similarité sémantique textuelle par rapport à un repaire déjà existant **[53]**. Dans notre travail ce modèle est celui de SemEval-2017 qui propose de mesurer le degré d'équivalence entre des extraits de texte appariés. L'objectif est de prédire la probabilité de similarité des deux extraits. Vu que notre vocabulaire est restreint ; notre objectif n'est pas de s'attaquer pleinement à cette tâche, mais bien de démontrer qu'en utilisant notre modèle, nous pouvons obtenir des scores de base raisonnables pour un tel tâche.

Afin de réaliser la comparaison on doit construire un vecteur pour tous les extraits en prenant la moyenne des vecteurs pour les mots dans le texte après avoir multiplié chaque vecteur par sa valeur TF – IDF (Term Frequency - Inverse Document Frequency)**[54]**. Ensuite nous allons calculer la similarité cosinus entre les vecteurs de chacun des deux extraits pour estimer la probabilité de similarité textuelle

Finalement nous allons utiliser l'outil d'évaluation officiel pour évaluer chaque modèle, comme indiqué dans le tableau suivant.

Les résultats obtenus montrent que l'application de cette approche sur notre model donne des valeurs moyennes, autrement dit l'efficacité de notre modèle est acceptables.

Model	Score
Model moyen	0.52033
Notre model	0.50628

Tableau 3.9 Score du model par rapport au score moyen des modèles de Sam Eval 2017

4.3 Ou se situe notre model par rapport aux autres modèles

La comparaison se fait essentiellement par rapport au model Aravec3.0

4.3.1 Inconvénients

*Le vocabulaire (nombre de mots incorporés) limité vu qu'il est spécifique à un domaine précis (l'hôtellerie), par rapport au vocabulaire du Aravec 3.0 qui est plus globale (Twitter, World Wide Web, Wikipédia), cet inconvénient peut affecter l'efficacité du model qui se montre très limité quand on l'applique dans d'autres domaines.

*Le problème que pose l'utilisation du dialecte : un mot peut avoir plus de sens selon la région

4.3.2 Avantages

Le vocabulaire spécifique du domaine de l'hôtellerie rend le model plus efficace [**55]** si on l'utilise dans les différentes taches de TALN notamment l'analyse de sentiments (tableau 3.3)

5. Conclusion

Notre model Word embedding a été réalisé en utilisant un corpus dont le contenu a été récupéré par la technique du web scraping, d'un site d'hôtellerie en limitant la récupération (les avis rédigés en arabe) .La langue arabe nécessite des prétraitements spéciaux vu sa morphologie et son lexique, ces prétraitements peuvent affecter l'intégrité du corpus ce qui nous impose des mesures de vérifications, ensuite on a construit le model Word embedding en ajustant les hyper paramètres afin d'optimiser l'efficacité du model.

Le model réalisé s'est montré efficace dans la détection des similarités, non similarité, du regroupement et la classification des groupes de mots.

Conclusion Générale

Nous avons abordé dans ce travail l'un des sujets en actualités dans le traitement automatique des langages naturels, celui de l'Analyse des Sentiments. Ce domaine reste toujours inexploité pour la langue arabe.

Dans ce contexte la technique du Word Embedding qui permet la représentation abstraite du vocabulaire et le traitement sémantique du langage est considérée comme une étapes très cruciale.

La construction d'un model Word Embedding pour un domaine spécifique (l'hôtellerie) de la langue arabe en prise en considération la spécificité de cette langue et les contraintes imposées représente un travail très utile.

Après la réalisation du notre model Word Embedding, il sera prêt pour l'exploitation dans le sens de :

- Analyse des Sentiments.
- Traitement Automatique des Langages Naturels
- Effectuer certaines analyses statistiques telles que la distribution de fréquence, la cooccurrence de mots…etc.
- Définition et validation des règles linguistiques pour diverses applications de TALN.
- Construction d'un système de correction de la grammaire.
- Définition des règles linguistiques spécifiques qui dépendent de l'utilisation de la langue.

Le model Word Embedding peut être amélioré par :

- Enrichissement du vocabulaire avec plus de données.
- Amélioration de la qualité du vocabulaire en appliquant d'autre types de prétraitement notamment pour la l'arabe dialecte.
- Création de plus de relations sémantique.
- Optimisation des valeurs des hyper paramètres du model.

Références bibliographiques

1 Dave, K., Lawrence, S., and Pennock, D. M. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. Proceedings of the World Wide Web Conference.2003

2 Bing Liu, Sentiment analysis and subjectivity. 2010

3 OEIL (OUTILS D'EXPLORATION ET D'INTÉGRATION DE LA LECTURE), Textes informatifs et textes d'opinion.

4 Sentiment analysis algorithms and applications: Walaa Medhat, Ahmed Hassan, Hoda Korashy ELECTRICAL ENGINEERING May 2014

5 Traitement Automatique du Langage Naturel (TALN) Outils d'analyse de données textuelles Laurent Audibert Université Paris 13 – novembre 2010

6 Bing Liu: Sentiment Analysis and Opinion Mining

7 Tang H, Tan S, and Cheng X. A survey on sentiment detection of reviews. Expert Systems with Applications: An International Journal, September 2009

8 Sébastien GILLOT, mémoire de master, Fouille d'opinions. Juin 2010

9 Gabriel Dabi-Schwebel, Microblogage " microblogging". Mai 2018.

10 Walaa MEDHAT, Ahmed HASSAN, Hoda KORASHY, Sentiment analysis algorithms and applications. 2014

11 Haseena Rahmath P, Deep learning - Challenges and Applications, Dept. of Computer Science and Engineering, Al-Falah School of Engineering, Dhauj, Haryana, India, May 2017

12 LEARNING WHILE SEARCHING IN CONSTRAINT-SATISFACTION-PROBLEMS*Rina Dechter Cognitive Systems Laboratory, Computer Science Department. University of California, Los Angeles.2017

13 Deep Learning in Neural Networks: An Overview Juergen Schmidhuber 2014

14 Vedran Vukotic, Vincent Claveau et Christian Raymond, « Supervised and Unsupervised Methods in Sentiment Analysis »2015

15 Exploiting Embedding in Content-Based Recommender Systems Yanbo Huang 2012

16 Distributional approaches to word meanings Chris Potts, Ling 236/Psych 236c: Representations of meaning, Spring 2013

17 Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching Word Vectors with Subword Information. In Emnlp

18 Mikolov, Tomas; et al. (2013). "Efficient Estimation of Word Representations in Vector Space"

19 Etude sur les représentations continues de mots appliqués à la détection automatique des erreurs de reconnaissance de la parole Jan 2018

20 Gensim - Deep learning with word2vec https://rare-technologies.com/deep-learning-with-word2vec-and-gensim/

21 Word Embeddings : Bénéfices d'une évaluation qualitative Bénédicte Pierrejean Novembre 2017

22 AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP Abu Bakr Soliman, Kareem Eissa, Samhaa R. El-Beltagy1 2017

23 Mohamed Hédi Maâloul , Approche hybride pour le résumé automatique de textes. Application à la langue arabe, Thèse de doctorat en informatique, soutenue le 18 décembre 2012, Université Aix –Marseille.

24 Traitement Automatique des Langues et Recherche d'Information en langue arabe dans un domaine de spécialité (Siham Boulaknadel) 2015

25 https://journals.openedition.org/aldebaran/68
26 http://jargonf.org/wiki/glyphe
27 M. K. Saad and W. Ashour, "Arabic morphological tools for text mining," Corpora, vol.18, 2010.
28 N. A. Abdulla, M. Al-Ayyoub, and M. N. Al-Kabi, "An extended analytical study of
arabic sentiments," International Journal of Big Data Intelligence, 2014.
29 F. Lazhar and T. G. Yamina, "Identification of Opinions in Arabic Texts using Ontologies," Information Technology & Software Engineering, 2012.
30 M. A. Zahran, A. Magooda, A. Y. Mahgoub, H. Raafat, M. Rashwan and A. Atyia, "Word Representations in Vector Space and their Applications for Arabic," in Computational Linguistics and Intelligent Text Processing: 16th International Conference, CICLing 2015, Cairo, Egypt,2015, Proceedings.
31 Identification automatique de mots clés dans les textes arabes- Boubekeur Yassamina - Mémoire de Master en Informatique -Ingénierie du Logiciel 2016- Université de Djilali BOUNAÂMA Khemis Miliana
32 Clitiques-Stemmer : nouveau stemmer pour la langue Arabe I. ZEROUAL, A. LAKHOUAJA Laboratoire de Recherche en Informatique, Faculté des Sciences Université Mohammed Premier ,Oujda, Maroc.
33 Polyglot: DistributedWord Representations for Multilingual NLP , Rami Al-Rfou Bryan Perozzi ,Computer Science Dept. Stony Brook University Stony Brook, NY
34 Named Entity Recognition for Arabic Social Media ,Ayah Zirikly-Department of Computer Science , June, 2015 ,The George Washington University -Washington DC, USA
35 AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP ,Abu Bakr Soliman, Kareem Eissa, Samhaa R. El-Beltagy1 2017, 3rd International Conference on Arabic Computational Linguistics, November 2017, Dubai, United Arab Emirates
36 https://www.Python.org
37 https://radimrehurek.com/gensim/
38 https://irkernel.github.io/
39 Initiation au Deep Learning avec Google Colab | Moov AI https://moov.ai/fr/blog/deep-learning-avec-google- colab
40 Biber, Douglas; Conrad, Susan; Reppen, Randi (1998). Corpus Linguistics. Investigating Language Structure and Use. Cambridge: CUP.
41 Kennedy, Graeme (1998). An Introduction to Corpus Linguistics. London & New York: Longman
42 N. A. Abdulla, M. Al-Ayyoub, and M. N. Al-Kabi, "An extended analytical study of arabic sentiments," International Journal of Big Data Intelligence 1, vol. 1,2014.
43 F. Lazhar and T. G. Yamina, "Identification of Opinions in Arabic Texts using Ontologies," Information Technology & Software Engineering. 2012
44 F. S. Douzidia and G. Lapalme, "Lakhas, an Arabic summarization system," in Proceedings of DUC, 2004.
45 S. Khoja and R. Garside, "Stemming arabic text," Lancaster, UK, Computing Department, Lancaster University,1999.
46 M. K. Saad and W. Ashour, "Arabic morphological tools for text mining," Corpora, vol. 18, p. 19, 2010
47 Y. Bengio, R. Ducharme, P. Vincent and C. Janvin, "A Neural Probabilistic Language Model," J. Mach. Learn. Res, vol. 3,2003.

48 R. Collobert and J. Weston, “A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning,”in Proceedings of the 25th International Conference on Machine Learning, New York, NY, USA, 2008.

49 T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient estimation of word representations in vector space” -2013.

50 BERNIER-COLBORNE G. & DROUIN P. (2016). Evaluation of distributional semantic models: a holistic approach. In Proceedings of the 5th International Workshop on Computational Terminology, Osaka, Japan. (2016).

51 ASR F. T., WILLITS J. A. & JONES M. N. (2016). Comparing Predictive and Co-occurrence Based Models of Lexical Semantics Trained on Child-directed Speech. In Proceedings of the 37th Meeting of the Cognitive Science Society. Austin, Texas.

52 LEVY O. & GOLDBERG Y. (2014). Dependency-Based Word Embeddings. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore

53 CHIU B., CRICHTON G., KORHONEN A. & PYYSALO S. (2016). How to Train Good Word Embeddings for Biomedical NLP. In Proceedings of the 15th Workshop on Biomedical Natural Language Processing, Berlin, Germany.

54 Evaluation of word embeddings against cognitive processes: primed reaction times in lexical decision and naming tasks Jeremy Auguste1, Arnaud Rey2, Benoit Favre1 Aix-Marseille Université, CNRS -Marseille, France

55 William L. Hamilton, Jure Leskovec, and Dan Jurafsky.2016. Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics,Berlin, Germany.