REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ 20 AOÛT 1955 -SKIKDA

Faculté des Sciences

Département d'Informatique

Mémoire de fin d'étude en vue de l'obtention du diplôme

De Master en Informatique

Spécialité : Système d'Information Avancée et Application

(SIAA)

THEME

Combinaison emojis-texte dans une architecture Transformer pour la classification des sentiments dans les tweets arabes

Réalisé Par :

- HATEM Rayane
- MERABET Nesrine

Encadré Par :

- Dr. BOUGAMOUZA Fateh

Année Universitaire 2022 – 2023

Remerciement

Tous d'abord, nous tenons à remercier le bon Dieu de nous avoir accordé toute la détermination, la volonté et la force pour qu'on puisse réaliser ce modeste travail.

Nous remercions infiniment notre encadreur ***Dr. BOUGAMOUZA Fateh*** *pour ses conseils, sa patience, sa disponibilité et son soutien tout au long de cette période.*

Nous tenons à exprimer notre profonde gratitude et nos sincères remerciements aux

Membres de jury d'avoir accepté de juger notre travail et de l'avoir enrichi.
Toutes les personnes qui nous ont aidés, de près ou de loin, en particulier
Nous adressons aussi nos remerciements à tous nos enseignants qui ont veillé sur notre formation.

Nous exprimons notre profonde gratitude envers notre famille pour leur amour, leur soutien et leur encouragement constants tout au long de notre parcours.

Nous tenons à remercier nos amis pour leur amitié indéfectible, leur soutien inconditionnel et les moments précieux que nous avons partagés ensemble.

Dédicace

À nos chers parents, qui ont été notre soutien inconditionnel, notre source d'amour et de sacrifices tout au long de notre parcours universitaire.

À nos chères sœurs et frères et précieux amis, qui ont été là à chaque étape, nous encourageons et nous motivons sans relâche.

À nos formidables professeurs et mentors, qui ont partagé leur savoir et leurs précieux conseils, nous guidant sur le chemin de la réussite.

À toutes les personnes qui ont obtenu de près ou de loin à la réalisation de ce travail de fin d'études.

Nous dédions humblement les fruits de notre travail, symboles de notre dévouement, de notre persévérance et de notre passion, Dans l'espoir que ces résultats vous valent de la fierté et contribuent positivement à notre domaine d'étude.

Nous exprimons notre gratitude infinie envers tous ceux qui nous ont soutenus tout au long de cette aventure enrichissante

Abstract

Social media, especially microblogging platforms like Twitter, provide a quick and easy way for users to share their opinions and feelings.
This thesis tackles the problem of sentiment classification in Arabic tweets using a combination of text and émojis in Transformer architecture. The study problem is the lack of resources and systems suitable for sentiment analysis in Arabic compared to other languages. The proposed solution is to train a BERT Transformer model on annotated data using both plain tweet text and associated émojis. The results obtained show a significant improvement over existing models, demonstrating the effectiveness of the proposed approach for the accurate classification of sentiments in Arabic tweets.

Key words: NLP, Sentiment Analysis, Transformer, Emojis, Twitter, Arabic language.

Résumé

Les médias sociaux, en particulier les plateformes de microblogging comme Twitter, offrent un moyen rapide et facile aux utilisateurs pour partager leurs opinions et leurs émotions.
Ce mémoire aborde le problème de la classification des sentiments dans les tweets arabes en utilisant une combinaison de texte et d'émojis dans une architecture Transformer. Le problème d'étude est le manque de ressources et de systèmes adaptés à l'analyse de sentiment en arabe par rapport à d'autres langues. La solution proposée consiste à entraîner un modèle BERT Transformer sur des données annotées en utilisant à la fois le texte brut des tweets et les émojis associés. Les résultats obtenus montrent une amélioration significative par rapport aux modèles existants, démontrant l'efficacité de l'approche proposée pour la classification précise des sentiments dans les tweets arabes.

Mots clés: PNL, Analyse des Sentiments, Transformer, Emojis, Twitter, Langue arabe.

الملخص :
توفر وسائل التواصل الإجتماعي ، وخاصة منصات المدونات الصغيرة مثل Twitter، طريقة سريعة وسهلة للمستخدمين لمشاركة آرائهم ومشاعرهم.
تتناول هذه الرسالة مشكلة تصنيف المشاعر في التغريدات العربية باستخدام مزيج من النص والرموز التعبيرية في بنية Transformer.
مشكلة الدراسة هي قلة الموارد والأنظمة المناسبة لتحليل المشاعر باللغة العربية مقارنة باللغات الأخرى. كان الحل المقترح هو تدريب نموذج BERT Transformer على البيانات المشروحة باستخدام كل من نص التغريدة العادي والرموز التعبيرية المرتبطة. أظهرت النتائج تحسنًا كبيرًا مقارنة بالنماذج الحالية ، مما يدل على فعالية النهج المقترح للتصنيف الدقيق للمشاعر في التغريدات العربية.

الكلمات المفتاحية : معالجة اللغة العصبية ، تحليل المشاعر ، التحويل ، الرموز التعبيرية ، تويتر، اللغة العربية.

Table des matières :

Chapitre 02: L'apprentissage automatique et l'apprentissage profond

Chapitre 03: Contribution et expérimentation

Chapitre 04:Implémantation

Liste des tableaux :

Liste des figures :

Pour les chapitres 01 et 02

Pour le chapitre 03

Pour le chapitre 04

Introduction générale

Introduction générale

Dans un monde entièrement connectée, les réseaux sociaux jouent un rôle central dans la communication, permettant aux individus d'exprimer leurs opinions, leurs émotions et leurs sentiments. Avec l'explosion des données diffusées sur les plateformes de médias sociaux, il est devenu essentiel de développer des méthodes efficaces pour analyser les sentiments exprimés par les utilisateurs. Une approche innovante pour communiquer les sentiments dans les messages textuels est l'utilisation des émojis, qui sont devenus des éléments incontournables de la communication en ligne.

La croissance exponentielle des réseaux sociaux a créé un immense réservoir de données, offrant une opportunité sans précédent d'explorer les sentiments des utilisateurs à grande échelle. L'analyse des sentiments, qui consiste à déterminer les émotions et les attitudes exprimées dans un texte, est essentielle pour comprendre les opinions et les tendances dans différents domaines tels que la politique, le marketing ou la santé publique. Cependant, l'analyse des sentiments dans les langues autres que l'anglais, en particulier la langue arabe, présente des défis uniques en raison de ses caractéristiques spécifiques. Ces difficultés sont liées à la richesse de la langue, à sa structure complexe et à l'utilisation courante des dialectes. De plus, les émojis utilisés dans les messages en arabe peuvent différer de ceux utilisés dans d'autres langues, ce qui nécessite une adaptation et une compréhension spécifiques. Par conséquent, il est essentiel de développer des méthodes adaptées pour combiner le texte et les émojis afin d'obtenir une analyse précise des sentiments exprimés dans les réseaux sociaux en langue arabe.

L'objectif de cette étude est d'explorer l'utilisation des émojis dans l'analyse des sentiments en langue arabe sur les réseaux sociaux. Nous cherchons à développer des méthodes innovantes qui intègrent à la fois le texte et les émojis pour améliorer la précision de l'analyse des sentiments. En utilisant des techniques avancées de traitement du langage naturel et d'apprentissage automatique, nous fournissons des résultats fiables et pertinents pour mieux comprendre les émotions transmises par les utilisateurs arabophones sur les réseaux sociaux

Ce mémoire sera structuré en quatre chapitres qui aborderont différents aspects de la combinaison texte-émojis pour l'analyse des sentiments dans les réseaux sociaux en langue arabe.

Le premier chapitre de ce mémoire offre une vue d’ensemble des concepts clés de traitement du langage naturel (NLP) et de l’analyse des sentiments dans les réseaux sociaux. Nous abordons les différentes techniques utilisées pour extraire des informations émotionnelles (positives et négatives) à partir des textes courts (des tweets).

Le deuxième chapitre se concentre sur l’apprentissage automatique et l’apprentissage profond, en mettant en évidence l’importance de ce domaine pour la classification des sentiments. Nous examinons les techniques d’apprentissage automatique couramment utilisés ainsi que les réseaux de neurones profonds, en soulignant l’efficacité des architectures Transformer pour le traitement des données textuelles.

Le troisième chapitre présente notre contribution à travers une étude approfondie de la langue arabe et de l’utilisation de Twitter comme source de données. Nous discutons également les deux bases de données que nous avons utilisées pour entraîner et évaluer notre système de classification des sentiments. Ensuite nous présentons les résultats expérimentaux obtenus et analysons les performances de notre approche.

Enfin, le dernier chapitre décrit l’implémentation de notre système, en mettant en évidence.

Chapitre 01

NLP et L’analyse des sentiments dans les réseaux sociaux

1. Introduction

Le traitement de langage naturel (NLP) est une technologie qui permet d'analyser et de comprendre le langage humain. Dans le contexte des réseaux sociaux, le NLP est devenu un outil incontournable pour comprendre et analyser les conversations en ligne. Les réseaux sociaux sont devenus une plateforme de communication et de partage d'informations de plus en plus importante dans la vie quotidienne, ainsi que dans les affaires. La combinaison des réseaux sociaux et du NLP a ouvert des nouvelles possibilités pour comprendre les tendances, les sentiments et les opinions des utilisateurs en ligne. Ce chapitre explore l'importance du NLP dans l'analyse des réseaux sociaux et les applications de cette technologie dans le contexte des réseaux sociaux en général.

2. Les réseaux sociaux

2.1. Définition des réseaux sociaux

Un “réseau social” est un graphe dont les nœuds sont des individus ou organisations, connectés par des liens représentant une relation “sociale” : appartenance à la même famille, échange de messages, goûts communs, etc (Viennet, 2008).

Michel Forsé a défini les réseaux sociaux comme suit : « un réseau social est un ensemble de relations entre un ensemble d’acteurs. Cet ensemble peut être organisé (une entreprise, par exemple) ou non (comme un réseau d’amis) et ces relations peuvent être de nature fort diverse (pouvoir, échanges de cadeaux, conseil, etc.), spécialisées ou non, symétriques ou non. Les acteurs sont le plus souvent des individus, mais il peut aussi s’agir de ménages, d’associations, etc. » (Forsé, 2008).

Les sites de réseaux sociaux sont des services Web qui permettent aux individus de construire un profil public ou semi-public dans un système délimité, d'articuler une liste d'autres utilisateurs avec lesquels ils partagent une connexion, et voir et parcourir leur liste de connexions et celles établies par d'autres au sein du système. La nature et la nomenclature de ces connexions peuvent varier d'un site à l'autre (Boyd, 2007).

Figure 1 :Un exemple d'organisation des relations dans un réseau social (Jarry, 2017)

2.2. Les caractéristiques des réseaux sociaux

Il est vrai que la classification des médias sociaux peut varier d'une personne à l'autre, mais certains caractères ont été généralisés dans les sous-plateformes de médias sociaux. Parmi eux : (Merzouk kahina, 2019)

- Multiplicité des plateformes, généralistes ou spécialisées (multiplication des applications).
- Plateformes adaptées et appropriées aux propres usages des utilisateurs.
- Les utilisateurs sont liés de façon bilatérale ou via des groupes : profils individuels, constitution de communautés, interaction avec le cercle des relations.
- Principe de cooptation et de recommandation.
- Gratuite et ouverture (pour la plupart des plateformes).

2.3. Les types des réseaux sociaux

Un réseau social est une plateforme qui permet à ses utilisateurs de créer des profils personnels pour communiquer et échanger des contenus divers tels que des photos, des vidéos, des articles de presse, des sites Internet, des opinions, des statuts, etc. Les réseaux sociaux offrent des services de messagerie et de discussion instantanée pour favoriser la communication entre les membres. Les réseaux sociaux ont un impact sur la communication en permettant aux membres de garder contact et alimentent un sentiment communautaire. Il existe plusieurs types de réseaux sociaux, certains généralistes comme Facebook, d'autres ont une portée professionnelle comme LinkedIn et Viadeo, et certains sont orientés autour d'une communauté particulière comme Spotify et Copains D'avant. Il existe même des réseaux sociaux dont l'accès est strictement limité à une communauté de privilégiés comme A Small World.

2.3.1. Les blogs

Une tendance à définir le blog comme un « journal personnel en ligne » (étienne, 2010). Un blog est un outil de publication en ligne qui permet à toute personne disposant d'une connexion Internet de partager ses idées et ses opinions avec le monde entier. Les ont permis à des millions de personnes de publier du contenu sur une grande variété de sujets. Les blogs sont également un excellent moyen pour les individus de se connecter avec d'autres personnes partageant les mêmes idées et de construire une communauté en ligne.

2.3.2. Microblogging

Le microblog est un dérivé du blog qui permet aux utilisateurs de publier de courts messages à l'attention de leur cercle de followers. Appelé à l'origine « tumblelog » ou journal de bord, il est renommé « microblog » vers 2006, au moment où le site de microblogging le plus célèbre, Twitter, est créé (Hassine). Comme pour les blogs, les interactions entre les microblogueurs et leurs followers sont alimentées par la publication de commentaires de ces derniers. Les fonctionnalités de ces plateformes restent relativement basiques, mais l'aspect relationnel est privilégié et la dimension communautaire est réduite à sa plus simple expression. Parmi les sites de microblogging les plus connus, on trouve bien sûr Twitter, qui permet aux utilisateurs de poster des tweets, c'est-à-dire des messages brefs limités à 280 caractères.

2.3.3. Sites de création et de partage de contenu

Ces plateformes en ligne permettent aux utilisateurs de créer et de publier du contenu de toutes sortes afin de le partager avec d'autres internautes et de recueillir leur avis, et cela représente l'une des manifestations les plus évidentes de l'UGC (en anglais : User Generated Content, en français : contenu généré par les utilisateurs). Chaque site de création et de partage de contenu a une vocation particulière, comme Flickr qui est consacré à la publication de photos ou YouTube qui est dédié au partage de vidéos.

Les projets collaboratifs permettent la création collective de contenu pour offrir un résultat de meilleure qualité qu'avec un seul individu. Les wikis sont l'exemple type de tels projets, permettant aux visiteurs de modifier le contenu des pages et l'encyclopédie collaborative Wikipedia étant le plus connu d'entre eux.

Les sites de social bookmarking, quant à eux, offrent aux utilisateurs la possibilité de partager, d'indexer et de recommander leurs liens préférés aux autres internautes. La viralité est l'une des caractéristiques de ce type de plateformes (Hassine).

2.3.4. Les forums

Les forums sont parmi les plus anciennes formes de médias sociaux, ayant été développés dès les débuts d'Internet. Ils permettent aux utilisateurs de participer à des discussions thématiques en ligne en ouvrant des sujets de conversation pour poser des questions, partager des informations ou recueillir des recommandations. Les forums sont une source importante de contenu généré par les utilisateurs. Doctissimo, un forum dédié à la santé, rassemble l'une des communautés les plus actives d'Internet en France (Hassine).

2.3.5. Les mondes virtuels

Les mondes virtuels sont des plateformes qui recréent un univers virtuel fantastique ou proche de la réalité, où les utilisateurs interagissent sous forme d'avatar. Bien qu'ils soient similaires aux jeux vidéo, ils nécessitent une connexion Internet et offrent également des services de discussion instantanée et des forums (Hassine). World of Warcraft et Second Life sont parmi les mondes virtuels les plus connus.

2.3.6. Les services de géolocalisation

Grâce à la popularité croissante des smartphones ces dernières années, de nombreuses applications de géolocalisation ont vu le jour sur ces appareils. Ces applications permettent aux utilisateurs de partager leur position avec leurs contacts, créant ainsi des opportunités pour établir des rencontres physiques entre les individus.

Par exemple, en utilisant la fonctionnalité de géolocalisation de Foursquare, deux personnes se trouvant dans le même endroit au même moment peuvent décider de se rencontrer. Ce type de fonctionnalité encourage les rencontres en face à face.

2.4. Les avantages et les inconvénients des réseaux sociaux

Les réseaux sociaux sont devenus omniprésents dans notre vie quotidienne, offrant des avantages tels que la connectivité et le partage de contenu, mais présentant également des inconvénients tels que la perte de confidentialité et la désinformation. nous allons approfondir l'examen des avantages et des inconvénients des médias sociaux pour mieux comprendre comment les utiliser de manière responsable et efficace.

2.4.1. Les avantages

- ***La communication*** : Les réseaux sociaux ont permis une communication facile et rapide entre les individus, quel que soit leur emplacement géographique.
- ***La connectivité*** : Les réseaux sociaux ont permis de connecter les personnes à travers le monde, en créant des communautés de personnes partageant des intérêts communs.
- ***Le partage d'informations*** : Les réseaux sociaux ont facilité le partage d'informations et de connaissances entre les utilisateurs, ce qui a permis d'améliorer l'accès à l'information.
- ***Le marketing*** : Les réseaux sociaux ont offert une plateforme pour les entreprises pour promouvoir leurs produits et services, et atteindre un public plus large.

- ***La mobilisation sociale*** : Les réseaux sociaux ont permis la mobilisation sociale et la sensibilisation à des questions importantes, ce qui a conduit à des changements sociaux positifs.
- ***L'expression personnelle* :** Les réseaux sociaux ont permis aux individus d'exprimer leur créativité et leur individualité, en partageant des photos, des vidéos et des pensées personnelles.

2.4.2. Les inconvénients

- ***La dépendance*** : Les réseaux sociaux peuvent devenir une habitude addictive qui peut affecter la santé mentale et le bien-être des utilisateurs.
- ***La désinformation*** : Les réseaux sociaux sont souvent utilisés pour diffuser des fausses informations et des théories du complot, ce qui peut affecter la perception du public sur les événements et les questions importantes.
- ***Le cyber intimidation*** : Les réseaux sociaux peuvent faciliter l'intimidation en ligne, en particulier chez les jeunes qui peuvent être victimes de harcèlement et de discrimination.
- ***La vie privée*** : Les réseaux sociaux peuvent compromettre la vie privée des utilisateurs en recueillant des données personnelles et en les partageant avec des tiers.
- ***Les effets sur la santé mentale*** : Les réseaux sociaux peuvent avoir un impact négatif sur la santé mentale des utilisateurs en augmentant le stress, l'anxiété, la dépression et l'isolement social.
- ***La manipulation de l'opinion publique*** : Les réseaux sociaux peuvent être utilisés pour manipuler l'opinion publique, en particulier lors des campagnes électorales et des événements politiques.

2.5. Twitter

Twitter est un réseau social de microblogage Il permet à un utilisateur d'envoyer gratuitement de brefs messages, appelés tweets, les gens utilisent des acronymes, commissent des erreurs d'orthographe, utilisent des émoticônes et d'autres caractéristiques qui expriment des significations particulières. Twitter est actuellement l'un des plates-formes de microblogage les plus populaires. Plusieurs célébrités utilisent Twitter, on y trouve même des chefs d'Etat.

C'est pourquoi de nombreux développeurs et de data scientiste utilisent les corpus de tweets pour leurs diversité et leurs richesse d'information textuelles (Mouhoubi azzedine mounir, 2020)

À l'origine, les tweets étaient limités à 140 caractères, mais le 7 novembre 2017, cette limite était doublée (280 caractères) pour toutes les langues sauf le japonais, le coréen et le chinois (Rosen, 2017).Voici une figure qui représente le logo de twitter :

Figure 2 : Logo de twitter **(Maybach, 2019)**

2.5.1. Les concepts de base Twitter

Le fonctionnement de ce réseau social est facile à comprendre. Un compte peut représenter diverses entités comme une personne, une entreprise ou un département. Les utilisateurs disposant d'un compte peuvent effectuer différentes actions telles que : (Goncalves, 2013)

- ***Hashtags*** : Les hashtags (#) sont des mots-clés précédés du symbole # qui aident à organiser et à trouver des contenus sur un sujet spécifique. Les utilisateurs de Twitter peuvent suivre des hashtags pour voir les tweets qui les utilisent.
- ***Mentions*** : Les mentions (@) sont des mentions de noms d'utilisateur Twitter dans un tweet. Les mentions permettent aux utilisateurs de se connecter et d'interagir avec d'autres utilisateurs.
- ***Retweets*** : Les retweets sont des messages partagés par les utilisateurs sur Twitter. Les retweets sont utilisés pour partager des tweets intéressants ou utiles avec d'autres utilisateurs.
- ***Favoris*** **:** Les favoris sont des tweets que les utilisateurs de Twitter marquent pour les lire plus tard ou les retrouver plus facilement.
- ***Abonnements*** **:** Les abonnements sont un moyen pour les utilisateurs de Twitter de suivre les tweets et les activités d'autres utilisateurs. Les abonnements permettent aux utilisateurs de suivre les tweets d'autres utilisateurs sans avoir à les suivre.
- ***Recherche*** **:** La recherche sur Twitter permet de trouver des tweets, des utilisateurs, des hashtags et des sujets pertinents pour un utilisateur.
- ***Statistiques*** : Twitter fournit des statistiques pour les utilisateurs, qui permettent de mesurer l'engagement, les impressions et les mentions de leurs tweets.
- ***Publicité*** **:** Les annonceurs peuvent utiliser Twitter pour promouvoir leur marque, leur produit ou leur service en utilisant les tweets sponsorisés ou les publicités Twitter.
- ***Direct Messages*** **:** Les messages directs sont des messages privés envoyés entre deux utilisateurs de Twitter. Les utilisateurs peuvent envoyer des messages directs à tout utilisateur qui suit leur compte Twitter.

3. Le traitement de langage naturel (NLP)

Le traitement de langage naturel est une application avancée de l'Intelligence artificielle et de l'apprentissage automatique utilisée pour comprendre le langage humain et pour extraire des informations sémantiques de toutes les sources des données qu'elles soient textuelles, audio ou vidéo.

3.1. Définition

Le traitement du langage naturel (NLP) est un domaine de l'informatique, de l'ingénierie et de l'intelligence artificielle qui permet aux machines d'interagir avec le langage naturel créé par les humains. Il utilise des techniques pour permettre aux ordinateurs de traiter et de comprendre le langage naturel humain, ainsi que pour fournir des résultats utiles (Jun Zhao, 2016).

3.2. Le processus de traitement de langage naturel (NLP)

Le traitement automatique du langage naturel (NLP) est une discipline qui vise à aider les ordinateurs à comprendre et à interagir avec le langage humain, que ce soit sous forme de texte ou de voix. Cette discipline peut être divisée en deux grandes parties : la compréhension du langage naturel et la génération du langage naturel (Barbosa, 2010)

3.2.1. La compréhension du langage naturel

La compréhension du langage naturel consiste à extraire des informations sémantiques à partir de données textuelles ou vocales. Elle utilise des techniques telles que la reconnaissance des entités nommées, l'analyse syntaxique, l'analyse sémantique et la reconnaissance de la langue pour permettre aux ordinateurs de comprendre le sens des phrases et des textes.

3.2.2. La génération du langage naturel

La génération du langage naturel, quant à elle, vise à produire du texte ou de la voix en utilisant des données d'entrée telles que des instructions ou des données structurées. Elle utilise des techniques telles que la planification de texte, la génération de texte et la synthèse vocale pour créer du contenu qui a l'apparence d'avoir été créé par un humain.

Ces deux parties du NLP sont importantes pour créer des applications pratiques telles que les chatbots, la traduction automatique, l'analyse de sentiments et la reconnaissance vocale, entre autres.

3.3. Les domaines d'application

Le traitement du langage naturel (NLP) a récemment attiré beaucoup d'attention sur l'analyse du langage humain, couvrants divers domaines tels que la traduction automatique, la détection de courrier électronique indésirable, les questions médicales, les réponses aux questions, le traitement et la synthèse de textes en langage naturel, la reconnaissance vocale, l'extraction d'informations, l'analyse de sentiment. (Khurana, 2023) (Collobert, 2008) (Djerrad, 2021)

3.3.1. Traduction automatique

Le développement d'algorithmes de traduction automatique a réellement révolutionné la manière dont les textes sont traduits aujourd'hui. Des applications, telles que Google Translator, sont capables de traduire des textes entiers sans aucune intervention humaine. Le langage naturel étant par nature ambigu et variable, ces applications ne reposent pas sur un travail de remplacement mot à mot, mais nécessitent une véritable analyse et modélisation de texte, connue sous le nom de Traduction automatique statistique (Statistical Machine Translation en anglais). (DataScientest)

3.3.2. Catégorisation de texte

Les systèmes de classification intègrent de grandes quantités de données, telles que des documents officiels, des rapports sur les pertes militaires, des données de marché, des fils de presse, etc. Puis ils les affectent à des catégories ou des index prédéfinis. Par exemple, le système Built de Carnegie Group capture les articles de Reuters et permet de gagner beaucoup de temps en effectuant un travail qui aurait nécessité du personnel ou des indexeurs humains. Certaines entreprises utilisent déjà des systèmes de triage pour catégoriser les tickets d'incident ou les demandes de réclamation et les acheminer vers le bureau approprié (Djerrad, 2021).

3.3.3. Le filtrage des spam

Le filtrage des spams fonctionne en utilisant la catégorisation de texte. Ces derniers temps, divers techniques d'apprentissages automatiques ont été appliqués à la catégorisation de texte, ou au filtrage anti-spam comme Rule Learning, Naïve Bayes, Machines à vecteurs de support, Arbres de décision etc. L'utilisation de ces approches est la meilleure car le classifier est appris à partir des données apprentissage plutôt que de le faire à la main. (Djerrad, 2021)

3.3.4. Résumé

L'ère numérique dans laquelle nous vivons engendre une surcharge d'informations qui dépasse largement notre capacité de compréhension. Cette tendance ne semble pas s'essouffler, il devient donc crucial de pouvoir résumer les données tout en conservant leur sens. Cette compétence est primordiale, non seulement pour identifier et donner du sens aux informations importantes dans un grand volume de données, mais aussi pour saisir leur signification émotionnelle profonde. Par exemple, une entreprise peut déterminer le sentiment dominant sur les réseaux sociaux et l'utiliser pour améliorer son produit, ce qui en fait un atout marketing précieux.

3.3.5. Système de dialogue

Les systèmes de dialogue sont limités aux niveaux phonétiques et lexicaux du langage et sont conçus pour des applications spécifiques, comme les réfrigérateurs ou les systèmes de cinéma maison. Cependant, en utilisant tous les niveaux de traitement du langage, ces systèmes de dialogue pourraient permettre aux robots d'interagir avec les humains dans leur langue naturelle, que ce soit par SMS ou par voix. Cette avancée ouvrirait de nouvelles possibilités pour l'interaction entre l'homme et la machine. Par exemple, l'assistant de Google, Windows Cortana, Siri d'Apple et Alexa, Amazon, sont les logiciels et les appareils qui suivent les systèmes de dialogue.

3.3.6. Extraction des informations

L'extraction de données est une tâche qui s'est développée dans le domaine de Traitement Automatique des Langues (TAL). Elle consiste à identifier et extraire d'un texte les éléments pertinents contenant des informations dont la nature est spécifiée à l'avance. Elle vise donc à transformer un texte de son format initial (une suite de chaines de caractères) à une représentation structurée et donc un format qui soit compréhensible par l'ordinateur. Elle se fait en reconnaissant dans le texte des unités lexicales particulières. (Bouaziz A. , 2013)

3.3.7. Analyse des sentiments

Aussi connue sous le nom de « Opinion Mining », l'analyse des sentiments consiste à identifier les informations subjectives d'un texte pour extraire l'opinion de l'auteur. À titre exemple, lorsqu'une marque lance un nouveau produit, elle peut exploiter les commentaires recueillis sur les réseaux sociaux pour identifier le sentiment positif ou négatif globalement partagé par les clients. De manière générale, l'analyse des sentiments permet de mesurer le niveau de satisfaction des clients vis-à-vis des produits ou services fournis par une entreprise ou un organisme. Elle peut même s'avérer bien plus efficace que des méthodes classiques comme les sondages. En effet, si l'on rechigne souvent à passer du temps à compléter de longs questionnaires, une partie croissante des consommateurs partage aujourd'hui fréquemment leurs opinions sur les réseaux sociaux. Ainsi, la recherche de textes négatifs et l'identification des principales plaintes permettent d'améliorer les produits, d'adapter la publicité et de réduire le niveau d'insatisfaction des clients (DataScientest).

3.4. Les principales méthodes en NLP

Globalement, nous pouvons distinguer deux aspects essentiels à tout problème de NLP. La partie linguistique permet de transformer le texte brut en données structurées, tandis que la partie d'apprentissage automatique permet de construire des modèles de Machine Learning qui peuvent utiliser ces données pour résoudre des problèmes spécifiques. (DataScientest)

3.4.1. La partie « linguistique »

Implique la compréhension de la langue naturelle et la façon dont les mots et les phrases sont construits et utilisés. Elle comprend des tâches telles que la segmentation de texte en phrases ou en mots, l'étiquetage de parties du discours, l'analyse syntaxique, la reconnaissance d'entités nommées et l'analyse sémantique. Ces tâches permettent de transformer le texte brut en données structurées que les algorithmes de Machine Learning peuvent utiliser pour la résolution de problèmes.

3.4.2. La partie « apprentissage automatique »

Implique la sélection des modèles de Machine Learning ou Deep Learning les plus appropriés pour la tâche spécifique à résoudre. Les modèles de NLP peuvent être supervisés ou non supervisés, et peuvent inclure des réseaux de neurones, des arbres de décision, des algorithmes de classification, des modèles de langue et bien d'autres encore. Une fois les

modèles sélectionnés, il est important de les entraîner sur des données annotées et de les évaluer pour mesurer leur performance et leur précision.

3.5. Les principaux modèles en NLP

Les modèles de traitement du langage naturel (NLP) ont connu des avancées significatives ces dernières années, permettant des avancées majeures dans des domaines tels que la traduction automatique, la génération de texte et l'analyse de sentiment. Ces avancées ont été soutenues par des entreprises technologiques de premier plan, notamment les géants du numérique GAFAM (Google, Apple, Facebook, Amazon, Microsoft), qui ont investi massivement dans la recherche et le développement de ces modèles. Ces derniers utilisent des réseaux de neurones profonds pour comprendre et générer du langage humain de manière sophistiquée.

Dans cette comparaison, nous allons examiner plusieurs modèles populaires de NLP, en mettant l'accent sur leurs caractéristiques clés. Le tableau ci-dessous présente une vue d'ensemble comparative des modèles suivants : GPT-1, BERT, GPT-2, RoBERTa, DistilBERT, et GPT-4 (López, 2023)¨ (DistilBERT) (LSTM, Transformers, GPT, BERT : guide des principales techniques en NLP) (Frąckiewicz, 2023)

Modèle de Langage Général	**Développeur**	**Année de sortie**	**Nombre de paramètres**	**Avantages**	**Inconvénients**
GPT-1	OpenAI	2018	117 millions	-capable d'effectuer un large éventail de taches linguistique et répondre aux questions.	-risque de biais : à cause de grand volume de corpus. -manque de « bon sens » -Interprétabilité limitée : A cause de grand volume de corpus.
BERT	Google	2018	110 millions	Amélioration des performances sur une gamme de taches NLP grâce à sa capacité d'apprendre des représentations contextuelles riche des mots.	Il partage avec GPT-1 certains inconvénients comme le manque de « bon sens » et Interprétabilité limitée.
GPT-2	OpenAI	2019	1.5 milliard	Beaucoup plus large que GPT-1 cette taille permet à GPT-2 de générer des textes de meilleure qualité et réaliser plus grand nombre de tâches comme le résumé et la traduction automatique.	Formation et déploiement difficile sur certain matériel à cause de ses exigences en ressource de calcul.
RoBERTa	Facebook	2019	125 millions	Amélioration des performances sur un large éventail de tache NLP comme la réponse aux questions, l'analyse des sentiments et la classification de texte en s'entrainant avec la phrase entière, plutôt qu'avec un seul masqué.	- Une lente inférence à cause de grand nombre de paramètres –le model est plus performant en anglais, mais pas le même dans les autres langues.

DistillBERT	Hugging Face	2019	Environ 66 millions	Version allégé de BERT, capable de réduire la taille du model de 40% et d'accélérer les temps d'inférence de 60% par rapport à BERT sans sacrifier la précision de ses performances.	-il n'est pas performant que BERT dans certains taches en NLP. -sa capacité de transférer ses connaissances à de nouvelles taches est limitée.
GPT-4	OpenAI	2023	Non déclarée	-Capable de traiter des images et des textes en entrée et produire des textes en sortie -un model plus puissant et sophistiqué que les autres modèles.	pourrait menacer les emplois des travailleurs humains. En outre, il pourrait également être utilisé pour générer des contenus malveillants, tels que de fausses nouvelles.

Tableau 1 : Vue d'ensemble comparative des modèles de NLP

4. L'analyse des sentiments

En raison de l'augmentation rapide du volume de textes subjectifs disponibles sur le web, tels que les blogs, les commentaires dans les forums, etc., les entreprises cherchent de plus en plus à obtenir des informations plus subtiles et subjectives - notamment des opinions - sur leurs produits en explorant internet.

4.1. Définition d'analyse des sentiments

L'analyse du sentiment (AS) aussi appelée opinion mining, est une branche relativement récente dans le domaine du *Natural Language Processing* (abrégé NLP), ou le Traitement Automatique du Langage Naturel (TALN). Cette discipline est généralement considérée comme une sous-catégorie de la classification de texte (Birjali, Kasri , & Ben-Hssane, 2021). Initialement connue sous le nom de Forage d'opinions. L'objectif de l'AS est d'extraire une classification subjective à partir d'un texte quelconque. D'un point de vue sémiotique, il s'agit essentiellement d'effectuer une analyse syntagmatique au moyen d'une approche informatique. De façon concrète, l'AS impliqué l'utilisation d'un modèle mathématique et statistique, dont les modèles d'apprentissage automatique, permettant à un programme informatique de «lire» un texte, de tel sorte à lui attribuer une classe subjective de sentiment ou un score de polarité, soit positif (1), négatif (-1) ou neutre (0).

4.1.1. Définition d'opinion

L'opinion est un avis, un jugement personnel que l'on s'est forgé sur une question ou un sujet en discussions qui ne relève pas de la connaissance rationnelle. L'opinion est aussi une manière de penser, un ensemble d'idées ou une doctrine (La Toupie).

4.1.2. Définition du sentiment

Le sentiment est le jugement que porte un individu sur un objet ou un sujet, ce jugement étant caractérisé par une polarité et une intensité. Tel qu'une polarité est soit positive, soit négative ou bien un mélange de ces deux valeurs, tandis que l'intensité montre le degré de positivité ou de négativité, et varie de faible à forte.

De cette définition, il ressort qu'un sentiment est un type particulier d'opinion dotée d'une polarité. Ainsi nous opposons les sentiments aux faits et aux expressions de neutralité face à un objet ou un sujet particulier (Pak, 2012).

4.2. Les niveaux d'analyse des sentiments

L'analyse des sentiments peut être réalisée à différents niveaux pour obtenir une compréhension approfondie des sentiments impliqués. Voici quelques niveaux d'analyse couramment utilisés (Kolkur, Gayatri, & Reena , 2015):

4.2.1. Au niveau du document

Détermine l'opinion générale de l'ensemble du document. Cette analyse fonctionne bien pour des documents qui présentent un point de vue précis, mais moins pour des comparaisons car elle ne fera pas la différence entre les sujets abordés.

4.2.2. Aux niveaux de la phrase

L'analyse des sentiments au niveau de la phrase est l'analyse la plus fine du document. En cela, la polarité est calculée pour chaque phrase car chaque phrase est considérée comme une unité distincte et chaque phrase peut avoir un avis différent.

4.2.3. Au niveau des aspects

Appelé en anglais Feature level: L'ingénierie des caractéristiques est une tâche extrêmement fondamentale et essentielle pour l'analyse de sentiment. L'étape de base dans l'analyse de

sentiment au niveau des caractéristiques est d'identifier le texte comme une caractéristique d'un produit. Par exemple, la durée de vie de la batterie est très longue. Dans cette revue, la batterie est une caractéristique du produit (nom) et "très longue durée de vie" est un mot d'opinion (adjectif).

4.3. Les approches de l'analyse des sentiments

Dans le domaine de la recherche, l'opinion mining est un sujet important qui a suscité de nombreuses approches à travers diverses études récentes. Pour illustrer ces méthodes, nous avons synthétisé certaines d'entre elles dans la figure suivante.

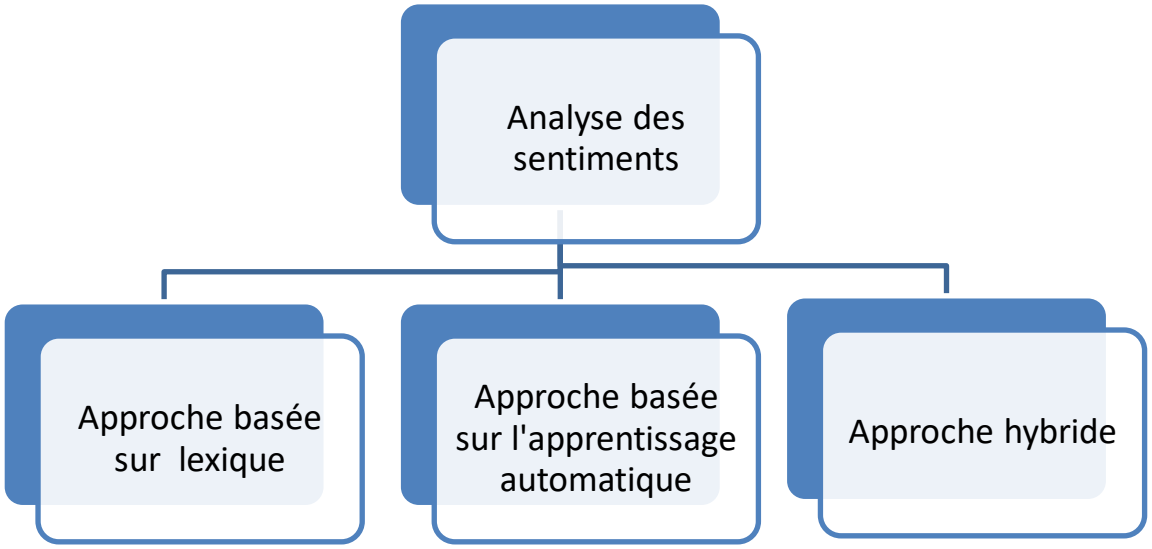


Figure 3 : Résumé des approches d'analyse des sentiments

4.3.1. Approche basée sur lexique

Elle consiste à identifier les mots-clés dans un texte pour déterminer l'opinion exprimée. Les mots sont généralement classés en trois catégories : positifs, négatifs et neutres. L'analyse lexicale peut être effectuée manuellement ou à l'aide de logiciels d'analyse de texte (Swarte, 2023).

4.3.2. Approche basée sur l'apprentissage automatique

Il utilise des algorithmes pour entraîner un modèle à identifier les sentiments dans un texte. Celui-ci est entraîné sur un ensemble de données d'apprentissage qui contient des exemples de textes avec des sentiments étiquetés (positifs, négatifs ou neutres). Il peut ensuite être utilisé pour prédire les sentiments dans de nouveaux textes (Swarte, 2023).

4.3.3. Approche hybride

Cette approche tire profit des deux méthodes précédentes. Il y a trois façons pour l'implémenter (Poirier, Fessant, Bothorel, Émilie , & Boullé, 2009):

- La première est d'exploiter les outils linguistiques pour élaborer le corpus puis classer les textes par un outil d'apprentissage supervisé.
- La deuxième façon est d'utiliser l'apprentissage automatique pour établir le corpus d'opinion nécessaire à l'approche basée sur lexique.

- La troisième façon est la combinaison des deux approches précédentes et le conjointement de leurs résultats.

4.4. Les domaines d'application d'analyse des sentiments

Au cours des dernières années, l'analyse de sentiment a suscité un grand intérêt en raison de ses nombreuses applications fascinantes dans divers domaines. Nous allons voir certaines de ces applications (Kharde & Sonawane, 2016).

4.4.1. Applications basées sur les avis de sites Web

Internet regorge aujourd'hui de critiques et de commentaires sur presque tout. Cela comprend les avis sur les produits, les commentaires sur les questions politiques, les commentaires sur les services, etc. Il est donc nécessaire d'avoir un système d'analyse de sentiment qui puisse extraire les sentiments sur un produit ou un service particulier. Cela nous aidera à automatiser la fourniture de commentaires ou de notes pour le produit ou l'article donné, ce qui répondra aux besoins des utilisateurs et des vendeurs.

4.4.2. Applications en tant que sous-composant technologique

Un système de prédiction de sentiment peut être utile dans les systèmes de recommandation. Le système de recommandation ne recommandera pas les éléments qui reçoivent beaucoup de commentaires négatifs ou de moins en moins de notes. Dans les communications en ligne, nous sommes confrontés à des langages abusifs et à d'autres éléments négatifs. Ces éléments peuvent être détectés simplement en identifiant un sentiment très négatif et en prenant des mesures correspondantes contre lui.

4.4.3. Applications en Business Intelligence

Il a été observé que les gens ont tendance de plus en plus à regarder les critiques de produits en ligne avant de les acheter. Et pour de nombreuses entreprises, l'opinion en ligne décide du succès ou de l'échec de leur produit. Ainsi, l'analyse de sentiment joue un rôle important dans les entreprises. Les entreprises souhaitent également extraire le sentiment des critiques en ligne afin d'améliorer leurs produits et, par conséquent, leur réputation et d'aider à la satisfaction de leurs clients.

4.4.4. Applications dans les maisons intelligentes

Les maisons intelligentes sont censées être la technologie de l'avenir. Dans le futur, toutes les maisons seraient en réseau et les gens pourraient contrôler n'importe quelle partie de la maison à l'aide d'un dispositif de tablette. Récemment, il y a eu beaucoup de recherches sur l'Internet des objets (Internet of Things abrégé IoT). L'analyse de sentiment trouvera également sa place dans l'IoT. Par exemple, en fonction du sentiment ou de l'émotion actuelle de l'utilisateur, la maison pourrait modifier son ambiance pour créer un environnement apaisant et paisible.

L'analyse de sentiment peut également être utilisée dans la prédiction des tendances. En suivant les opinions du public, des données importantes sur les tendances de vente et la satisfaction des clients peuvent être extraites.

4.5. L'analyse des sentiments et les émojis

4.5.1. Émoticône, emoji et Kaomoji

Il existe souvent une confusion entre émoticônes et émojis, les deux langages sont pourtant différents, et n'ont pas la même histoire.

- Les émoticônes : ils sont une représentation faciale faite à partir de caractères typographiques. Les plus connus sont évidemment les :-(ou :-). Il est communément admis qu'ils ont été inventés en 1982 par le chercheur en informatique Scott Fahlamn à l'université de Carnegie Mellon. Les émoticônes permettaient de différencier contenu sérieux et plus léger. Ils sont une contraction de « emotion icon ».

- Les Kaomoji : utilisés originellement au Japon, les Kaomojis sont la version japonaise des émoticônes et ils utilisent les spécificités des caractères et de la ponctuation japonaise. Si les émoticônes occidentaux mettent surtout en avant les expressions de la bouche, les Kaomojis sont eux plus portés sur les yeux. Précisons aussi qu'ils ont l'avantage de se lire de façon horizontale, contrairement aux émoticons. Voici quelques Kaomoji qui expriment la joie (◕‿◕) (o˘◡˘o), l'amour (ー‿‿ー)♡ (˘‿˘)♡(˘‿˘) (´ ▽ ` ).。o ♡ , la colère ( # `Д´) (` 皿´ #) ┌∩┐ (◣_◢) ┌∩┐ ou la peur {{ (>_<) }} |ı|(ºΔº)|ı|.

- Les émojis : inventés plus tardivement, les émojis sont nés en 1999 grâce à l'entreprise japonaise NTT DoCoMo. L'opérateur téléphonique était confronté à un problème qui semble aujourd'hui lointain, ses abonnés utilisaient de plus en plus de messages photos. Ces envois d'images consommaient énormément de données. Pour pallier à cela, DoCoMo a inventé l'emoji qui permet de transformer un émoticône en image et donc de consommer moins de données et moins de caractères par message. Depuis 2010, les émojis font partie du langage Unicode Standard, et ont connu un développement fulgurant, notamment grâce à Apple qui a intégré ce clavier Unicode emoji en 2011 à ses dispositifs. Il y a 972 caractères Emoji disponibles dans le langage Unicode 7.0 (Ropars, 2018).La figure 4 représente une liste d'émojis qui partagent des sentiments différant utilisées dans les réseaux sociaux.

Figure 4 : Liste des émojis qui partagent des sentiments différant **(Forum, 2022)**.

4.5.2. Les différentes dimensions des émojis

Comme l'explique le sociologue André Gunthert, les émojis donnent une valeur émotionnelle de plus au langage. Ils ont trois dimensions essentielles (Émoji : le nouveau langage des émotions):

- Esthétique : car un message avec des images est plus joli, c'est décoratif.
- Ludique : car cela apporte un degré de fantaisie au contenu.
- Sémiotique : une image peut signifier plusieurs choses et peut être interprétée de façon plus large qu'un message linguistique.

4.5.3. Utilisation et interprétation des émojis

Les émojis sont un moyen pour communiquer des significations extralinguistique très complexes. Lorsque nous dialoguons, nous exprimons ce commentaire métalinguistique avec un ton, une intonation ou un stress. Par exemple, je peux parler avec une voix sarcastique, mais je ne peux pas exprimer cette même voix par l'écrit. Cependant, je peux mettre un émoji avant ou après un texte pour montrer qu'il doit être compris d'une voix sarcastique. Ces symboles prennent en charge le langage utilisé par quelqu'un et le poussent encore plus loin. En effet il est difficile de communiquer nos sentiments par le verbal. Par exemple, lorsque une personne est triste, il serait plus simple d'utiliser un émoji pour l'exprimer, que d'en faire avec des mots, et cela donne aux rédacteurs la possibilité d'exprimer rapidement, pas seulement une émotion basique mais aussi de différents objets qu'on utilise dans notre vie courante, grâce aux différentes catégories des émoticônes (Rawlings, 2018). Lors d'une communication écrite, une phrase peut avoir un état émotionnel impartial. Cependant si on utilise un émoji on peut tout simplement ajouter une émotion ou refondre en totalité le sens de cette phrase, comme nous allons le démontrer dans l'exemple suivant :

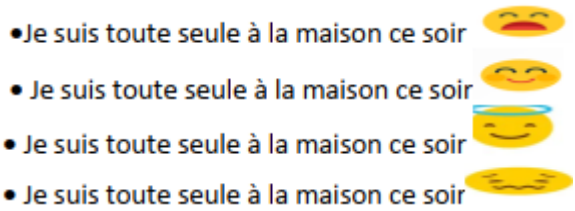


Figure 5: L'effet de l'ajoute des émojis sur le sens de la phrase **(Farida, 2022)**.

4.6. L'analyse des sentiments et la langue arabe

4.6.1. La langue arabe et ses caractéristiques

L'arabe est la langue officielle de plus de 20 pays avec plus de 375 millions de locuteurs natifs. Ceux-ci sont largement concentrés au Moyen-Orient, mais il existe des groupes minoritaires de locuteurs natifs à travers le monde. C'est également une langue officielle des Nations Unies, de l'Organisation de la conférence islamique et de l'Union africaine. En plus des locuteurs natifs, plusieurs millions d'autres connaissent l'arabe comme langue étrangère car, en tant que langue du Coran, il est compris par les musulmans du monde entier (arabe,

2023).

L'arabe est une langue sémitique (VERSTEEGH, 1997) et se compose de nombreux dialectes régionaux différents. Bien que ces dialectes soient de véritables formes de langue maternelle, ils sont généralement utilisés dans la communication quotidienne informelle et ne sont pas standardisés ou enseignés dans les écoles (Habash N. Y., 2010).

Il existe une norme écrite formellement utilisée dans les médias écrits et l'éducation dans le monde arabe, appelée Arabe Standard Moderne (ASM). L'ASM diffère fondamentalement de la plupart des dialectes arabes et, fait intéressant, elle n'est la langue maternelle d'aucun pays ou groupe arabe. L'arabe possède un système d'inflexion très riche et est considéré comme l'une des langues les plus riches en termes de morphologie (Habash, Rambow, & Roth, 2009). Dans la langue arabe, il existe deux constructions principales : nominales et verbales (Farra, Abou Assi, & M. Hajj, 2010). Dans le domaine verbal, l'arabe à deux modèles d'ordre des mots, c'est-à-dire sujet-verbe-objet et verbe-sujet-objet. Dans le domaine nominal, un modèle courant consiste en deux mots consécutifs, un nom (le sujet) suivi d'un adjectif.

4.6.2. La langue arabe dans les réseaux sociaux

Lorsqu'on examine rapidement les techniques de communication linguistique dans les réseaux sociaux, on constate une diversité linguistique déconcertante, associée à des limites linguistiques variables. Les utilisateurs d'aujourd'hui utilisent un langage complètement nouveau, transgressant les modèles linguistiques établis et exploitant les interdictions linguistiques pour exprimer leur créativité et leur individualité. Ainsi, la langue des réseaux sociaux est devenue une langue à part entière, à l'intérieur même de la langue principale.

Cette nouvelle langue comporte plusieurs éléments linguistiques qui interprètent la langue d'origine pour créer une deuxième, voire une troisième langue. Elle repousse les limites linguistiques en transformant les chiffres en lettres, en plus d'abuser de chevauchements entre différents éléments linguistiques tels que l'argot, l'arabe, le français, voire l'anglais. Cette langue influence également l'économie linguistique en mêlant lettres et chiffres pour des usages linguistiques spécifiques, tout en incorporant des langues vernaculaires et des dialectes souvent entremêlés avec leurs variantes associées (La Réalité de La Langue Arabe sur Les Réseaux Sociaux et Son Impact Sur La Sécurité Linguistique et L'identité, 2019).

4.6.3. Les défis de l’analyse des sentiments dans la langue arabe

L'analyse des sentiments pour la langue arabe présente plusieurs défis spécifiques. Voici quelques-uns des principaux défis rencontrés dans ce domaine :

La complexité de la langue arabe : L'arabe est une langue complexe avec une grammaire riche et des structures de phrases différentes de celles des langues occidentales. Les expressions idiomatiques et les nuances linguistiques peuvent rendre difficile l'interprétation précise des sentiments (Duwairi, Marji, & Rushaidat, 2014).

- Diversité linguistique : L'arabe est parlé dans de nombreux pays, et chaque région peut avoir ses propres dialectes et variantes linguistiques. La diversité linguistique rend difficile le développement de modèles d'analyse des sentiments qui fonctionnent de manière cohérente sur l'ensemble de la langue arabe.
- Manque de ressources annotées : Pour entraîner des modèles d'analyse des sentiments, il est nécessaire d'avoir des ensembles de données annotées avec des étiquettes de sentiment.

Cependant, il existe un manque de ressources annotées pour l'arabe, ce qui limite la disponibilité des données d'entraînement de haute qualité.

- Sensibilité culturelle : Les sentiments et les émotions sont souvent liés à la culture et à l'expérience individuelle. Les modèles d'analyse des sentiments développés pour d'autres langues peuvent ne pas être directement applicables à l'arabe en raison des différences culturelles. Il est donc important de tenir compte de la sensibilité culturelle lors de l'analyse des sentiments en arabe.
- Ironie et sarcasme : L'arabe est une langue connue pour son utilisation fréquente de l'ironie et du sarcasme. La détection précise de ces nuances linguistiques peut être un défi pour les modèles d'analyse des sentiments qui ne parviennent pas toujours à capturer ces subtilités.
- Traitement des émoticônes et des expressions informelles : Les conversations en ligne en arabe peuvent inclure l'utilisation fréquente d'émoticônes, d'argot et d'expressions informelles. Ces éléments ajoutent une complexité supplémentaire à l'analyse des sentiments, car ils obtiennent une compréhension approfondie du contexte et de la culture pour être interprétés correctement.

Ces défis soulignent l'importance de développer des modèles spécifiques à la langue arabe et de disposer de ressources et de données d'entraînement adéquates pour améliorer l'analyse des sentiments dans cette langue.

5. Travail connexe

L'analyse des sentiments dans la langue arabe avec les emojis présente des défis uniques, notamment en raison du manque de travaux connexes dans ce domaine spécifique.

- **Arabic Sentiment Analysis using Arabic-BERT**

Cet article se propose d'explorer l'analyse des sentiments en arabe en utilisant à la fois des modèles d'apprentissage automatique classiques et des techniques d'apprentissage profond. Tout d'abord, des modèles ML classiques sont entraînés sur le corpus arabic-sentiment-twitter-corpus. Ces modèles sont ensuite évalués sur trois autres bases de données arabes étiquetés en fonction des sentiments.

Ensuite, une approche d'apprentissage profonde est explorée en affinant un modèle Arabic-BERT. Ce modèle Arabic-BERT a été pré-entraîné sur un vaste corpus d'environ 8,2 milliards de mots. Plusieurs itérations sont effectuées, en expérimentant différentes variations de l'architecture du modèle. Les performances de ces modèles sont attribuées en utilisant les mêmes trois bases de données.

En plus de varier l'architecture du modèle, l'article examine également l'impact du changement du bases de données d'entraînement. Les trois bases données sont utilisés individuellement comme base de données d'entraînement, et l'article proposé comment ces changements offrent les performances globales du modèle ainsi que sa capacité à généraliser.

En combinant à la fois des modèles d'apprentissage automatique classiques et des techniques d'apprentissage profond, cet article vise à offrir une exploration complète de l'analyse des sentiments en arabe, en examinant différentes approches et en évaluant leur efficacité dans ce contexte spécifique. Cet article a résulté un modèle BERT-base de performance 0.899.

Ces travaux connexes fournissent une base solide pour notre étude et servent de référence aux différentes approches et méthodologies utilisées (HANY, Arabic Sentiment Analysis using Arabic-BERT, 2021).

6. Conclusion

En conclusion, l'utilisation de l'analyse des sentiments dans les réseaux sociaux a apporté une nouvelle dimension à l'analyse du langage naturel (NLP). Grâce aux progrès de l'apprentissage automatique et de l'intelligence artificielle, il est maintenant possible de

comprendre et d'interpréter les émotions exprimées par les utilisateurs des réseaux sociaux à grande échelle.

L'analyse des sentiments permet aux entreprises et aux organisations de tirer parti de la richesse des données générées sur les plateformes sociales. Elle leur permet de mesurer l'opinion publique, de comprendre les réactions des utilisateurs à leurs produits ou services, et même de prédire les tendances émergentes. Les marques peuvent utiliser ces informations pour adapter leurs stratégies marketing, améliorer leurs relations avec les clients et prendre des décisions plus éclairées.

De plus, l'analyse des sentiments joue également un rôle crucial dans le domaine du service client sur les réseaux sociaux. En identifiant les messages négatifs ou les plaintes des clients, les entreprises peuvent intervenir rapidement pour résoudre les problèmes et éviter les retombées négatives. De même, l'identification des messages positifs permet de détecter les clients satisfaits et de renforcer leur engagement envers la marque

Chapitre 02

L’apprentissage Automatique et l’apprentissage profond

1. Introduction

L'apprentissage automatique et l'apprentissage profond sont deux domaines de l'intelligence artificielle qui ont connu une croissance rapide ces dernières années. Ils permettent aux ordinateurs de traiter et d'analyser des données de manière autonome, en apprenant à partir d'exemples plutôt que d'être explicitement programmés.

L'apprentissage automatique est utilisé dans une grande variété de domaines, tels que la reconnaissance de la parole et de l'image, la prédiction de séries chronologiques, la recommandation de produits, et bien d'autres. L'apprentissage profond, quant à lui, utilise des réseaux de neurones artificiels pour résoudre des problèmes complexes, tels que la reconnaissance faciale, la conduite autonome, le traitement automatique des sentiments, et la traduction automatique.

Ces technologies ont le potentiel de transformer de nombreux domaines, y compris la médecine, la finance, l'industrie manufacturière, l'agriculture, et bien d'autres. Elles permettent de prendre des décisions plus éclairées, de prédire les tendances futures, et d'automatiser des tâches qui étaient auparavant réservées aux humains.

Cependant, l'apprentissage automatique et l'apprentissage profond soulèvent également des questions éthiques et de sécurité, telles que la protection de la vie privée et la fiabilité des décisions prises par les machines. Il est donc important de comprendre ces technologies et leurs implications pour notre société.

2. L'apprentissage automatique

L'apprentissage automatique (ML Machine Learning en anglais) est une branche de l'IA qui permet aux systèmes d'apprendre et de s'améliorer à partir de l'expérience sans être explicitement programmés. L'apprentissage automatique se concentre sur le développement de programmes informatiques capables d'accéder aux données et de les utiliser pour apprendre par eux- mêmes (What Is Machine Learning? A Definition, 2022).

On distingue ainsi plusieurs types d’apprentissage : l'apprentissage supervisé, l’apprentissage non supervisé, l’apprentissage par renforcement, l’apprentissage par transfert et la figure ci-dessous résume ces types.

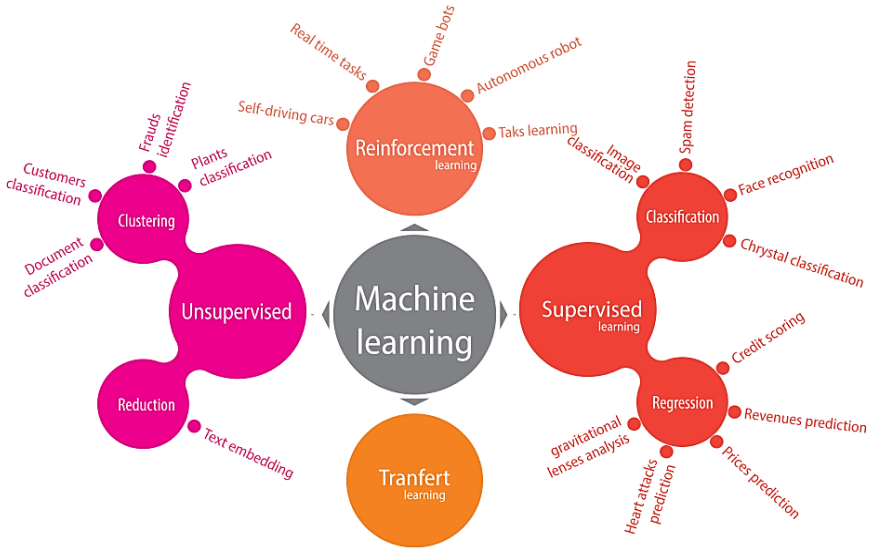


Figure 6: Les différents types d'apprentissage automatique

2.1. L'apprentissage supervisé

L'apprentissage supervisé est une sous-catégorie de l'apprentissage automatique et de l'intelligence artificielle. Il est défini par son utilisation d'ensembles de données étiquetés pour former des algorithmes qui permettent de classer les données ou de prédire les résultats avec précision. L'objectif est de donner un sens aux données dans le contexte d'une question spécifique. L'apprentissage supervisé est utilisé pour des problèmes de classification et de régression, comme la détermination de la catégorie à laquelle appartient un article de prcesse, ou la prévision du volume des ventes pour une date future donnée (Belaidi, 2022).

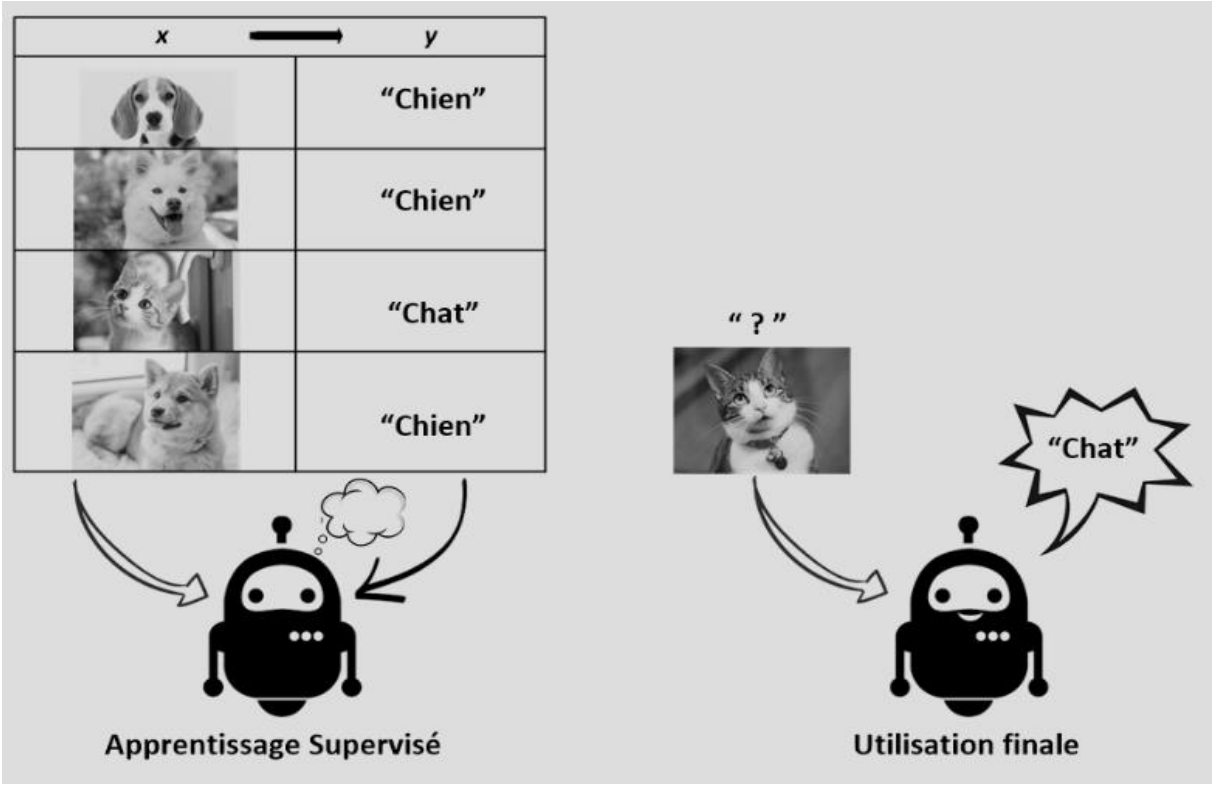


Figure 7 : Un exemple d'apprentissage supervisé (Saint-Cirgue, 2019).

2.2. L'apprentissage non supervisé

Au contraire, si seul des exemples sans étiquettes sont disponibles, et si les classes et leurs nombre sont inconnus, on parle d'apprentissage non supervise. Dans ce cas, l'apprentissage se ramène alors à cibler les groupes homogènes d'exemples existants dans les données, c'est-à-dire à identifier des groupes tels que les exemples les plus similaires appartiennent au même groupe, et que les exemples les plus différents soient séparés dans différents groupes, la notion de similarité étant le plus souvent ramenée à une fonction de distance entre paires d'exemples. Autrement dit, il s'agit à ce niveau de rechercher la distribution sous-jacente des exemples dans leur espace de description. Voici un exemple d'apprentissage supervisé (Abdelhamid, 2021).

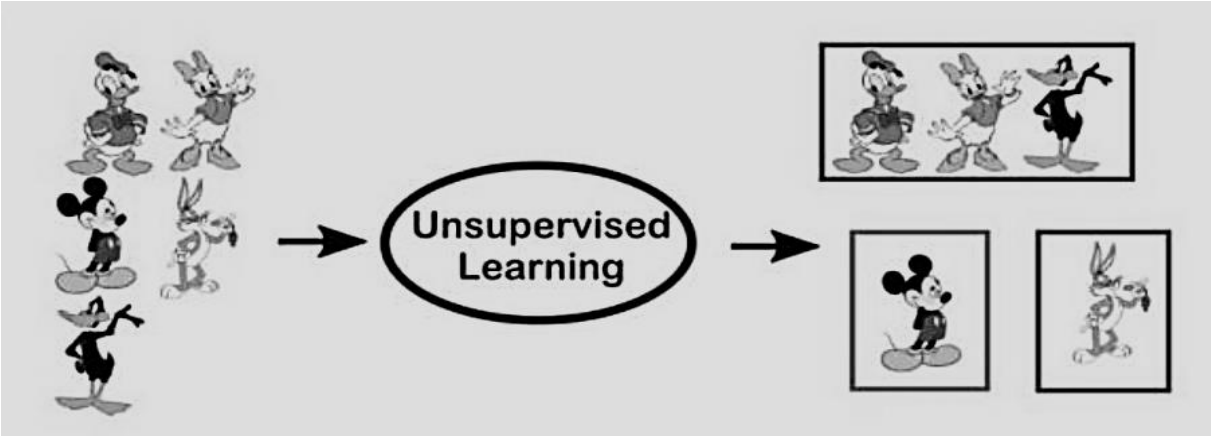


Figure 8 Un exemple d'apprentissage non supervisé **(Ismaili, 2019)**.

2.3. L'apprentissage par renforcement

Avec l'apprentissage par renforcement la machine n'a pas besoin de l'aide de l'être humain, ni en termes de supervision, ni en termes de fourniture de données. L'apprentissage par renforcement est une branche très différente. Le système d'apprentissage, appelé un agent dans ce contexte (Voir figure 8), peut observer l'environnement, sélectionner et effectuer des actions, et enfin obtenir des récompenses ou des pénalités (des récompenses négatives). La machine peut apprendre toute seule la meilleure stratégie à suivre, appelée une politique, pour obtenir plusieurs récompenses au fil du temps. Une politique définit l'action que l'agent devrait choisir lorsqu'il est dans une situation donnée (Géron, 2019). Voici un exemple d'apprentissage par renforcement.

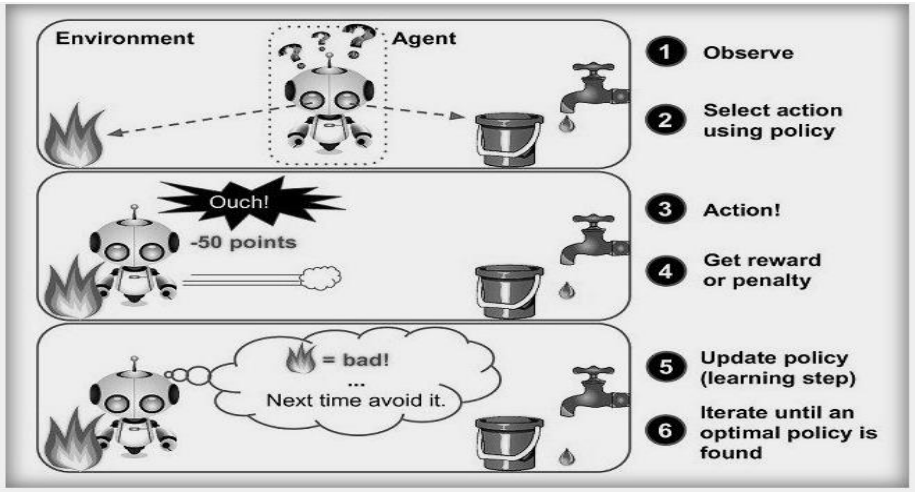


Figure 9 : Un exemple d'apprentissage par renforcement (Team, 2020).

2.4. L'apprentissage par transfert

L'apprentissage par transfert (transfer learning, en anglais) consiste à appliquer des connaissances obtenues en effectuant une tâche afin de résoudre un problème différent, mais qui présente des similitudes.

L'apprentissage par transfert permet d'exploiter les connaissances tirées d'une tâche source pour améliorer l'apprentissage dans l'exécution d'une nouvelle tâche. Si la méthode de transfert entraîne une baisse des performances de la nouvelle tâche, on parle de transfert négatif. Lorsqu'on élabore une méthode de transfert, l'un des principaux enjeux consiste à créer un transfert positif entre des tâches similaires, mais en évitant un transfert négatif entre des tâches moins similaires. Lorsqu'on applique à une nouvelle tâche des connaissances apprises d'une tâche source, les caractéristiques de cette dernière sont généralement mises en correspondance avec celles de la nouvelle tâche (Apprentissage par transfert).

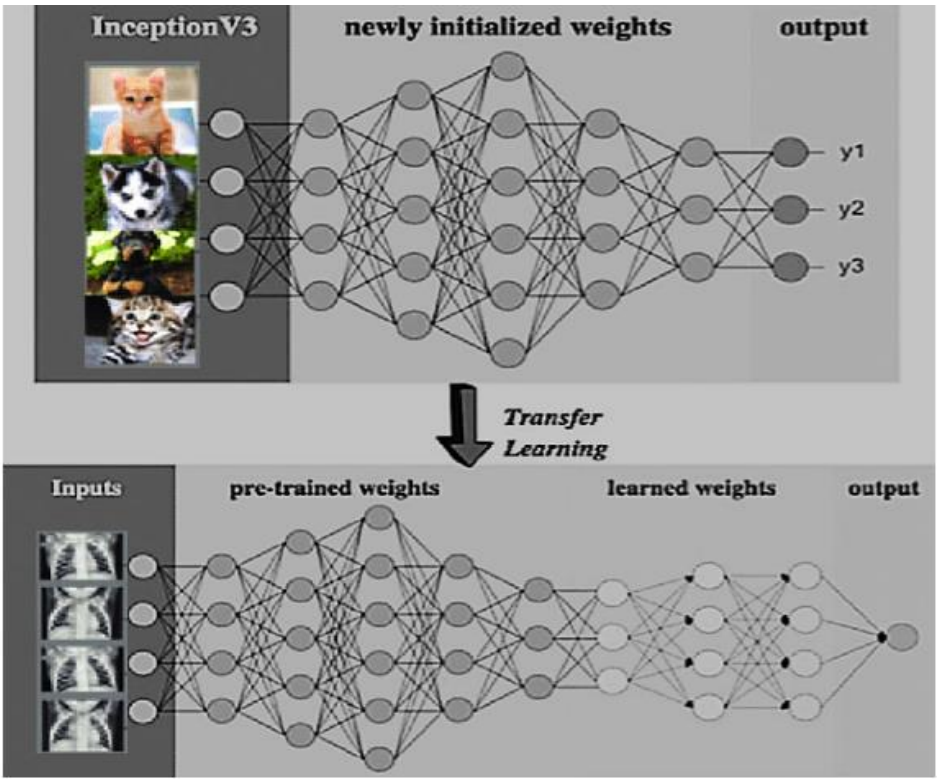


Figure 10 :Un exemple d'apprentissage par transfert (Mohammed, Karrar , Begonya , Salama, & Mashael , 2020)

3. L'apprentissage profond

Le terme « apprentissage profond » a été introduit dans le domaine du l'apprentissage automatique par « Rina Dechter » en 1986 (Dechter, 1986), et dans les réseaux de neurones artificiels par «Yann Lecun» en 1989, dans le contexte des réseaux neurones convolutifs. L'apprentissage profond (DL Deep Learning en anglais) désigne une technique d'apprentissage automatique (voir figure 11) qui vise à construire automatiquement des connaissances à partir de grandes quantités d'information. Les caractéristiques essentielles du traitement ne seront plus identifiées par un traitement humain dans algorithme préalable, mais directement par l'algorithme d'apprentissage profond (Deep Learning : définition, concept et

usages potentiels, 2017).

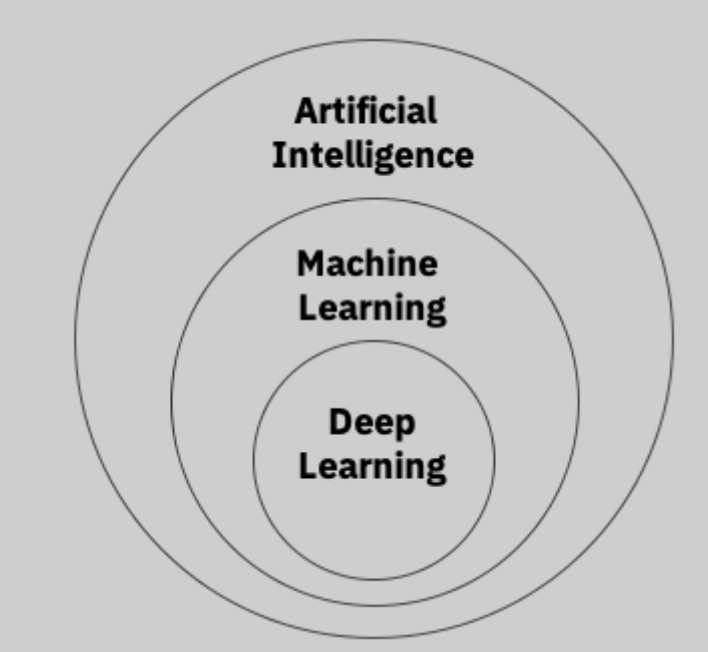


Figure 11: la relation entre IA et ML et DL **(Moore, 2021)**

L'apprentissage profond permet donc implicitement de répondre à des questions du type «que peut-on déduire de ces données ?» et décrire des caractéristiques parfois cachées ou des relations entre des données souvent impossibles à identifier pour l'homme. D'après (Patterson, 2017) l'apprentissage profond est un réseau neuronal avec un grand nombre de paramètres et de couches.

3.1. Domaine d'application

La technologie d'apprentissage en profondeur est l'une des techniques les plus utilisées dans de nombreux domaines, notamment :

- ***la reconnaissance faciale*** : une application de deep learning, capable d'apprendre à identifier un visage ou détecter des caractéristiques comme les yeux, le nez et la bouche sur une photo.
- ***Le traitement automatique de langage*** **naturel** : une autre application du deep learning, qui vise à extraire le sens des mots, voire des phrases.
- ***voiture autonome*** **:** Les entreprises qui développent des services d'assistance à la conduite doivent enseigner à un ordinateur à maîtriser des aspects clés de la conduite en utilisant des systèmes de capteurs numériques plutôt que le jugement humain. Pour ce faire, ces entreprises commencent généralement par entraîner des algorithmes en utilisant de vastes ensembles de données (Mittal, 2017).
- ***Assistants vocaux*** : C'est également le cas des assistants vocaux, tels que Siri, Alexa ou Google Home. Ceux-ci se fondent sur la technologie du deep learning pour développer leur compréhension du langage et leur vocabulaire. Tout comme les chatbots qui permettent de répondre de plus en plus précisément aux diverses demandes des clients (Welter, 2023).

- ***la détection du cancer du cerveau*** : les algorithmes de deep learning permettent de détecter les cellules cancéreuses plus facilement et de réduire le cancer résiduel après l'opération (Mittal, 2017).
- ***Marketing*** : Par ailleurs, l'apprentissage profond trouve toute sa place dans le marketing automation. Il facilite l'élaboration de campagnes publicitaires et d'e-mails ultra personnalisés. Il peut aussi servir à optimiser le score des leads, à classer et à faire remonter les problèmes des clients (Welter, 2023).

4. Les réseaux de neurones

4.1. Les neurones biologiques

Un neurone biologique est composés d'un corps cellulaire, d'axones, de dendrites et de synapses, sont capables de traiter et de transmettre l'activation neuronale.

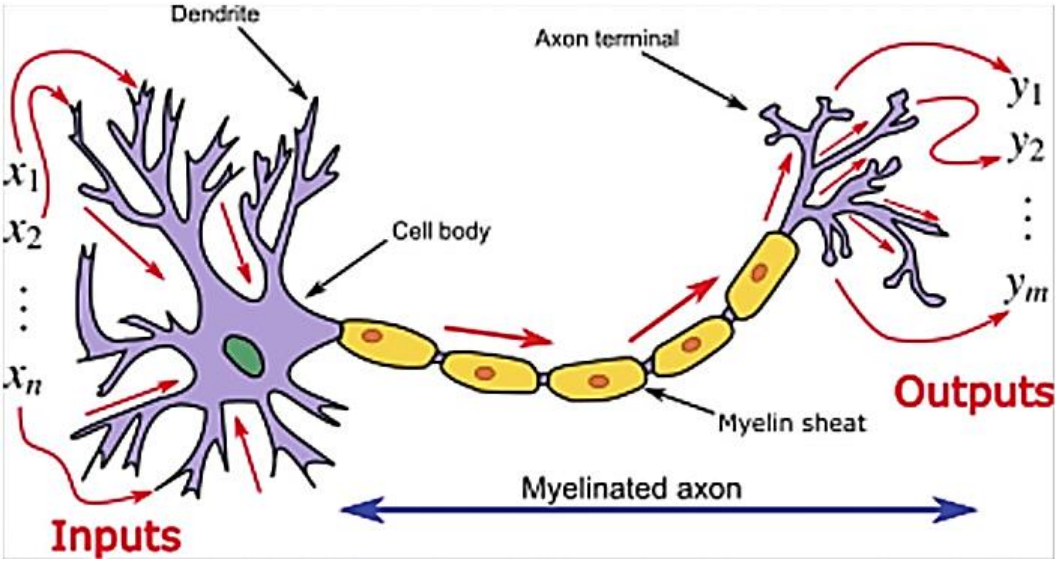


Figure 12 : Un neurone biologique **(Rj, 2020)**

4.2. Le neurone artificiel

Le neurone artificiel est un modèle de calcul dont la conception est inspirée par le fonctionnement des neurones biologiques. Ce neurone formel peut être considéré comme un opérateur recevant un nombre variable d'entrées de l'environnement externe ou d'autres neurones, chacun de ces entrées est pondéré par le poids appelé poids synaptique, et fournit une sortie seulement lorsque la somme dépasse un certain seuil interne (Bouaziz & Khantoul, 2020). La figure 13 montre un neurone artificiel. L'Interconnexion de ces neurones produit un réseau de neurones.

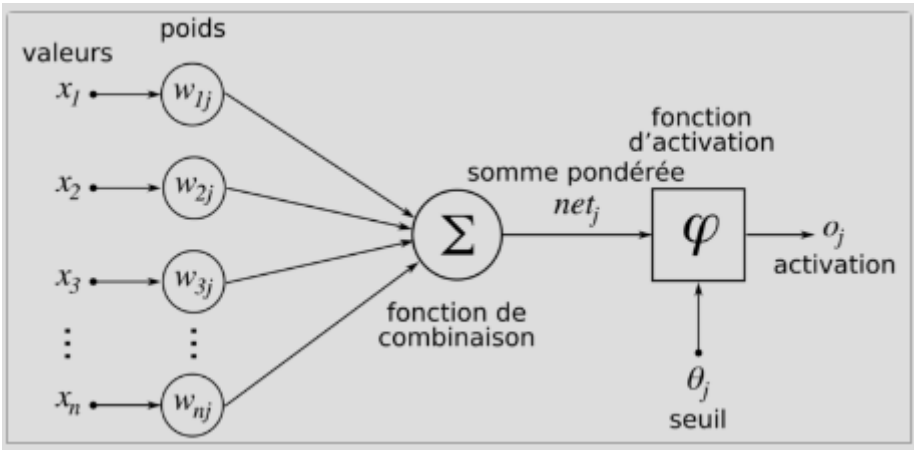


Figure 13 Structure d'un neurone artificiel **(Kaadoud, 2018)**

4.3. La règle de Hebb

C'est une technique d'apprentissage qui permet de modifier les poids des connexions entre les neurones. La variation introduite dans les poids des connexions entre deux neurones a et b est formalisés par l'équation :

$$\Delta_{Wab} = \alpha . y_a . x_b$$

Où Wab représente le poids reliant les neurones a et b, Δ_{Wab} est la variation de Wab, y_a est la sortie du neurone a, x_b représente l'entrée du neurone b et α est une constante modélisant la vitesse d'apprentissage (HEBB, 1949).

4.4. Le perceptron

Il s'agit d'un neurone artificiel conçu pour automatiser le classement de données. Il a été inventé en 1957 par Frank Rosenblatt, au sein du (Rosenblatt, 1957).

4.4.1. Le perceptron monocouche

Un perceptron simple ou monocouche (ou single-layer perceptron) est un réseau de neurones artificiels composé d'entrées multiples et d'une seule sortie. Le cheminement des informations se fait par un circuit de liens qui relient entre les entrées et les neurones regroupés sur une couche unique. Le perceptron simple fonctionne sur un modèle basique de type classifieur linéaire (qui sépare les données en deux classes). Il permet par exemple d'apprendre à identifier un spam à partir des mots détectés dans un e-mail. L'inconvénient du perceptron monocouche tient dans l'impossibilité de travailler sur des données complexes et en grand nombre. Le modèle d'apprentissage reste basique et limité dans ses applications, car la séparation des classes n'est effective que de manière linéaire (Crochet-Damais A. , 2022).

4.4.2. Le perceptron multicouche

Un perceptron multicouche (ou multilayer perceptron) est un réseau de neurones artificiels comprenant plusieurs couches qui permet de produire un séparateur non linéaire. Il est constitué de plusieurs entrées et sorties. Il s'agit d'un réseau à propagation directe (feedforward) de plusieurs couches disposant chacune d'un nombre de neurones artificiels variable. Ce type de modèle est

notamment utilisé dans la reconnaissance d'images ou la détection de fraude, par exemple (Crochet-Damais A. , 2022).

4.5. Les fonctions d'activation

Les fonctions d'activation sont essentiellement utilisées dans les réseaux neuronaux artificiels pour convertir un signal d'entrée en un signal de sortie, qui est ensuite utilisé comme entrée pour la couche suivante du réseau. Dans un réseau neuronal artificiel, on effectue d'abord le calcul de la somme pondérée des entrées en utilisant les poids associés, puis on applique une fonction d'activation à cette somme afin d'obtenir la sortie de cette couche spécifique. Cette sortie est ensuite transmise en tant qu'entrée à la couche suivante du réseau (Sharma, Sharma , & Anidhya , 2020).

4.6. Les types des fonctions d'activation

Il existe plusieurs types de fonction d'activation nous citons quelque exemples (Keldenich, 2021).

- ***La fonction de ReLU***

La fonction Rectified Linear Unit (**ReLU**) est la fonction d'activation la plus simple et la plus utilisée. Elle donne x si x est supérieur à 0**,** 0 sinon. Autrement dit, c'est le maximum entre x et 0 **:**

$$fonction\_ReLU(x) = \max(x, 0)$$

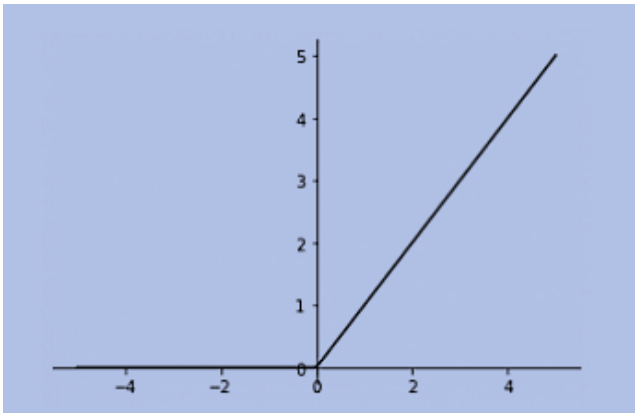


Figure 14 : Fonction ReLU **(Keldenich, 2021)**.

- ***La fonction de GeLU***

L'unité linéaire d'erreur gaussienne, ou GeLU, est une fonction qui multiplie simplement son entrée par la fonction de densité cumulée Φ de la distribution normale à cette entrée. Parce que ce calcul est plutôt lent, une approximation considérablement plus rapide qui ne varie que dans la quatrième décimale est fréquemment utilisée dans la pratique.

$$fonction\_GeLU = x\Phi(x)$$

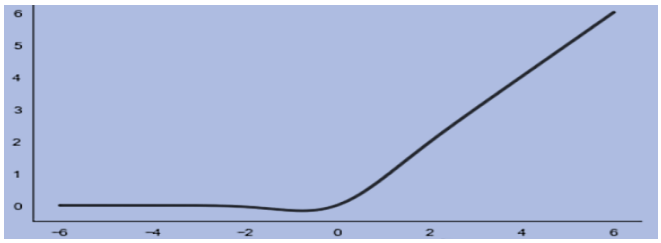


Figure 15 : Fonction GeLU (Shrivastav, 2023)

- ***La fonction de Sigmoïde***

La fonction Sigmoïde donne une valeur entre 0 et 1**,** une probabilité. Elle est donc

très utilisée pour la classification binaire, lorsqu'un modèle doit déterminer seulement deux labels. Ainsi, pour la classification des critiques de cinéma, plus la valeur retournée par Sigmoïde est proche de 1 plus le modèle considère que la critique est positive.

Au contraire, plus elle est proche de 0, plus elle est considérée comme négative :

$$fonction\_Sigmoide(x) = 1/(1 + \exp(-x))$$

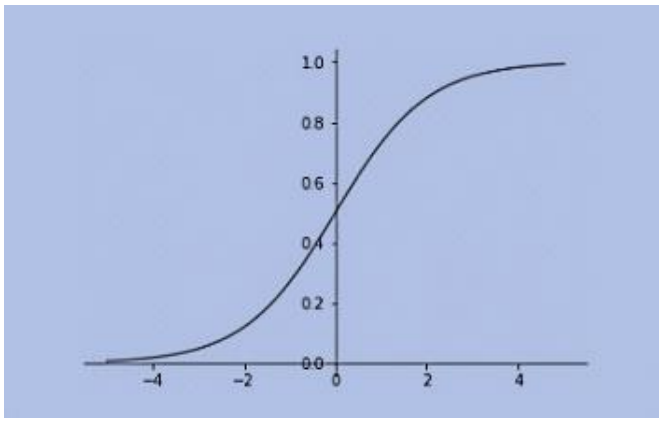


Figure 16 : Fonction Sigmoïde

- ***La fonction de Softmax***

La fonction Softmax permet de transformer un vecteur réel en vecteur de probabilité. On l'utilise souvent dans la couche finale d'un modèle de classification**,** notamment pour les problèmes multiclasse. Dans la fonction Softmax, chaque vecteur est traité indépendamment. L'argument axis définit l'axe d'entrée sur lequel la fonction est appliquée :

$$fonction\_Softmax(x) = \exp(x)/tf.reduce\_sum(\exp(x))$$

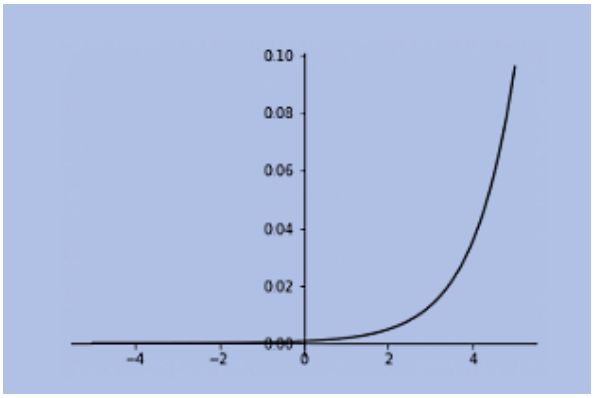


Figure 17 : Fonction Softmax

- ***La fonction de Tanh***

La fonction tanh est simplement la fonction de la tangente hyperbolique.

Il s'agit en fait d'une version mathématiquement décalée de la fonction sigmoïde :

sigmoïde donne un résultat entre 0 et 1

tanh donne un résultat entre -1 et 1

L'avantage de tanh est que les entrées négatives seront bien répertoriées comme négatives là où, avec sigmoïde, les entrées négatives peuvent être confondus avec les valeurs proche de nulles.

$$fonction\_\tanh(x) = \sinh(x)/\cosh(x)$$

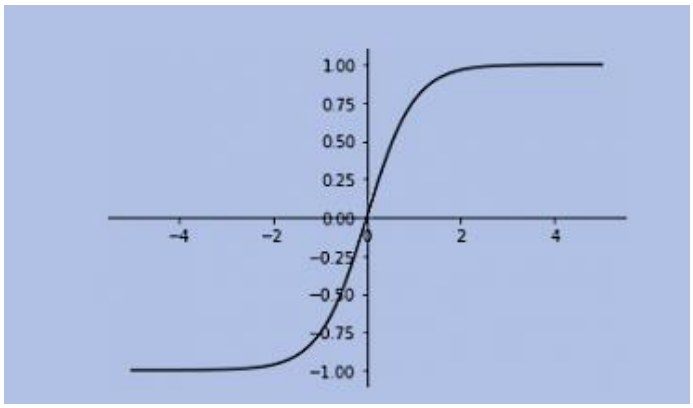


Figure 18 : Fonction tanh

4.7. Les types de réseaux de neurones

4.7.1. Réseau de neurones entièrement connectés (Fully connected)

Les réseaux de neurones entièrement connectés FCNN sont la base de nombreux algorithmes du Deep learning. Ils sont efficaces pour le traitement de tâches complexes à partir d'un grand

volume de data (big data). Ils sont également plus difficiles à entraîner qu'un autre réseau neural.

Un réseau de neurones complexe est un réseau neural comportant un grand nombre de couches cachées. Ici, « entièrement connecté » signifie que chaque neurone de la couche cachée est connecté à chaque neurone de la couche d'entrée.

Le réseau neural entièrement connecté est le modèle le plus courant de réseau neural complexe. Le fait d'être « entièrement connecté » (perceptron multicouche) permet aux informations de circuler librement entre tous les neurones du réseau. Le principal avantage de l'utilisation de ces couches entièrement connectées est qu'elles permettent aux réseaux profonds d'apprendre des relations complexes entre les data d'entrée et de sortie.

Les FCNNs peuvent également souffrir d'overfitting, et il convient donc de faire preuve de prudence lors de la conception d'un réseau neural profond. Ils ne sont pas connectés localement et ne peuvent donc pas tirer parti de certains modèles (tels que les modèles spatiaux).

Les réseaux entièrement connectés ne sont enfin pas très évolutifs : plus le nombre de codes augmente, plus le nombre de connexions (et donc la quantité de data d'apprentissage nécessaires) augmente de manière exponentielle (Turpin).

4.7.2. Réseau de neurones convolutifs (CNN)

Les réseaux de neurones convolutifs désignent une sous-catégorie des réseaux de neurones, ils sont spécialement conçus pour traiter des images en entrée. Leur architecture est alors plus spécifique : elle est composée de trois types de couches principales, qui sont les suivants (Réseaux neuronaux convolutifs):

- ***La couche convolutive*** : est un composant clé des réseaux neuronaux convolutifs (CNN). Elle effectue la majorité des calculs dans le réseau et nécessite des données d'entrée, un filtre et une carte de caractéristiques. L'entrée typique est une image en couleur représentée par une matrice de pixels en 3D, où chaque dimension correspond à la hauteur, la largeur et la profondeur (RVB) de l'image. Un filtre également appelé détecteur de caractéristiques ou noyau, est utilisé pour vérifier la présence de caractéristiques en se conduisant dans les champs réceptifs de l'image. Ce processus est connu sous le nom de convolution.

Le filtre est une matrice bidimensionnelle de poids qui représente une partie de l'image, généralement de taille 3x3. Il est appliqué à une zone de l'image, et un produit scalaire est calculé entre les pixels d'entrée et le filtre. Le résultat est ensuite stocké dans une matrice de sortie, appelée carte de caractéristiques, carte d'activation ou caractéristique convoluée. Le filtre se déplace ensuite d'un pas et le processus se répète jusqu'à ce que tout l'image soit balayée.

Après chaque opération de convolution, une transformation non linéaire appelée ReLU (Rectified Linear Unit) est appliquée à la carte de caractéristiques. Cela a introduit la non-linéarité dans le modèle et aide à capturer des informations plus complexes.

- ***Les couches de mise en commun ou de pooling*** : également appelées sous-échantillonnage, effectuent une réduction de la dimensionnalité, en réduisant le nombre de paramètres en entrée. Comme pour la couche convolutive, l'opération de mise en commun applique un filtre sur l'ensemble de l'entrée, mais la différence est que ce filtre ne comporte pas de poids. Au lieu de cela, le noyau applique une fonction d'agrégation aux valeurs situées dans le champ réceptif, remplissant ainsi le tableau de sortie

- ***Couche entièrement connectée* :** Cette couche effectue la tâche de classification sur la base des caractéristiques extraites par les couches précédentes et leurs différents filtres.

4.7.3. Réseau de neurones récurrents (RNN)

Les réseaux récurrents ou dynamiques ou feedback sont caractérisés par des connexions bouclées. Ils permettent alors la modélisation de système à mémoire interne ainsi que des systèmes variant dans le temps. Ainsi grâce à sa mémoire adaptable il a la capacité d'implémenter directement des systèmes dynamiques. Ces réseaux récurrents sont caractérisés par des cycles si on les représente sous forme de graphes. Ceci introduit une dimension temporelle dans son comportement. Ils se caractérisent par deux types de comportement dynamique :

- Une dynamique autonome convergente (pour des entrées fixées)
- Une dynamique non-autonome non-convergente (pour des entrées variantes dans le temps)

Le réseau de Hopfield et la machine de Boltzman rentrent dans la première catégorie. Les réseaux de la deuxième catégorie utilisent le contexte courant et leurs entrées variantes dans le temps pour évoluer suivant leur dynamique. (MORERE, 2021)

4.7.3.1. Architecture des RNN

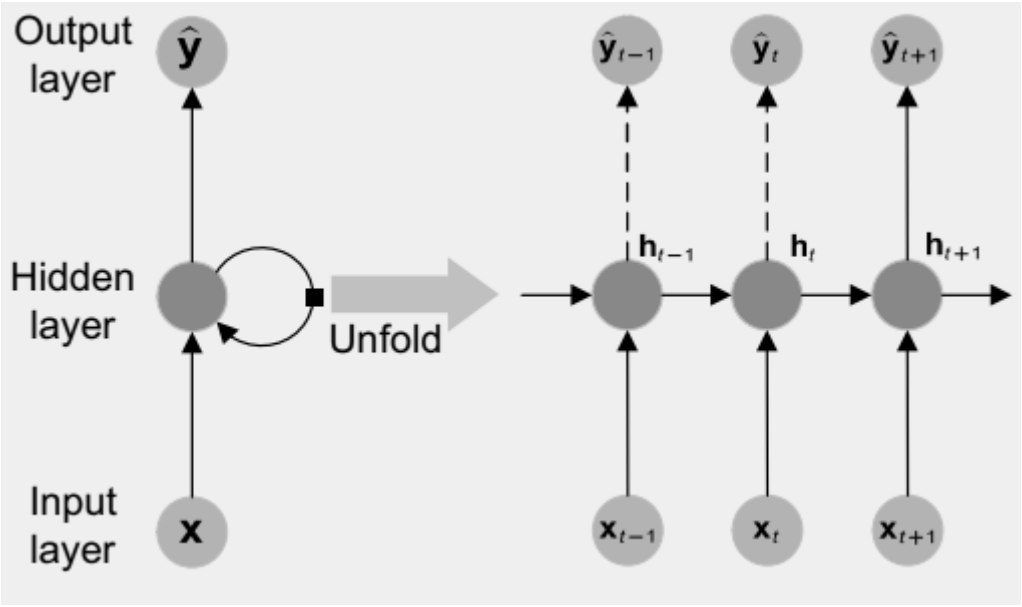


Figure 19 : Architecture du RNN **(Hua, Zhifeng , & Rongpeng , 2018)**

La figure 19 montre l'architecture du RNN (Réseau de Neurones à Mémoire à Court Terme). Étant donné une série temporelle d'entrée $x = \{x_1, x_2, \dots x_T\}$, le RNN calcule de manière itérative la séquence d'états cachés $h = \{h_1, h_2, \dots, h_T\}$, ainsi que la séquence de sorties

$y = \{y_1, y_2, \dots, y_T\}$, en utilisant l'ensemble d'équations suivant :

$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + b_h)$.

$y_t = g(W_{yh}h_t + b_h)$.

x_t est la valeur de la série temporelle à l'instant t.

h_t est l'état caché à l'instant t.

y_t est la sortie à l'instant t.

W_{hx} est le poids de connexion entre les entrées et l'état caché.

W_{hh} est le poids de connexion entre l'état caché précédent et l'état caché actuel.

W_{yh} est le poids de connexion entre l'état caché et la sortie.

b_h et by sont les biais associés respectivement à l'état caché et à la sortie.

f et g sont des fonctions d'activation non linéaires.

Ces équations décrivent comment le RNN traite séquentiellement les entrées temporelles et met à jour son état caché et ses sorties à chaque pas de temps, en utilisant les informations des pas de temps précédents. Cette récurrence dans le calcul des états cachés permet au RNN de capturer les dépendances à long terme dans les séries temporelles et de modéliser les relations séquentielles entre les différentes valeurs (Baheti, 2021).

4.7.3.2. Types de RNN (simple, LSTM, GRU)

Les réseaux de neurones traditionnels (simples) analysent des données étiquetés mais ne sont pas conçus pour faire des prédictions de séries temporelles (données évoluant dans le temps). Pour réaliser ce type de calcul, il existe trois grands types de réseaux de neurones récurrents : le RNN simple, le LSTM et enfin le GRU. Le RNN simple (appelé souvent « Vanilla RNN ») est la forme la plus basique de RNN. Il ne possède pas de portes (Gate), ce qui signifie que le flux d'informations n'est pas contrôlé et les informations essentielles à la tâche à accomplir peuvent être écrasés par des informations redondantes ou non pertinentes c'est le problème d'évanescence du gradient. En pratique, les RNN vanille ne sont pas utilisés et sont seulement étudiés à des fins d'enseignement (Hairy, 2021).

Pour surmonter ce problème, deux versions spécialisées de RNN ont été créées :

- ***LSTM (Long Short Term Memory).***

La mémoire à long terme en bref LSTM est un type spécial de RNN capable d'apprendre des

séquences à long terme. Ils ont été introduits par Schmidhuber *et Hochreiter* en 1997 (Hochreiter & Schmidhuber, 1997). Il est explicitement conçu pour éviter les problèmes de dépendance à long terme. Se souvenir des longues séquences pendant une longue période de temps est sa façon de travailler.

La popularité de LSTM est due au mécanisme d'obtention impliqué dans chaque cellule LSTM. Dans une cellule RNN normale, l'entrée à l'horodatage et à l'état caché de l'étape de temps précédente est passée à travers la couche d'activation pour obtenir un nouvel état. Alors que dans LSTM, le processus est légèrement complexe, comme vous pouvez le voir dans l'architecture ci-dessus à chaque fois qu'il prend l'entrée de trois états différents comme l'état d'entrée actuel, la mémoire à court terme de la cellule précédente et enfin la mémoire à long terme. Ces cellules utilisent les portes pour réguler les informations à conserver ou à rejeter lors du fonctionnement en boucle avant de transmettre les informations à long terme et à court terme à la cellule suivante. Nous pouvons imaginer ces portes comme des filtres qui suppriment les informations sélectionnées indésirables et non pertinentes. Il y a un total de trois portes que LSTM utilise comme : porte d'entrée, porte oubliée et porte de sortie (Lendave, 2021) .

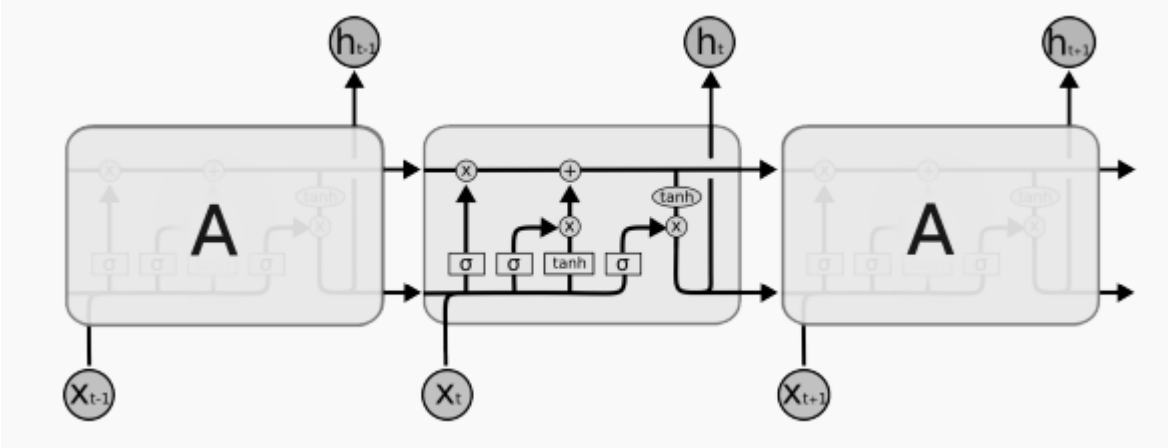


Figure 20 : Architecture d'une cellule LSTM **(Risser-Maroix & Kerboua-Benlarbi, 2018)**

- ***GRU (Gated Recurrent Units)***

Les GRU sont un type de réseau neuronal récurrent (RNN) qui utilise une structure plus simple que les LSTM et est plus facile à former. Ils ont deux portes : une porte de mise à jour et une porte de réinitialisation. La porte de mise à jour détermine les informations de l'état caché précédent et de l'entrée actuelle à conserver, et la porte de réinitialisation détermine les informations à rejeter. L'état caché final est une combinaison des informations retenues par la porte de mise à jour et de l'entrée actuelle. Cette combinaison de portes permet aux GRU de conserver les informations pertinentes à partir de longues séquences et de supprimer les informations non pertinentes ou obsolètes. Les GRU sont souvent utilisés pour des tâches impliquant des données séquentielles, telles que la traduction linguistique et la modélisation linguistique (Harsha, 2023).

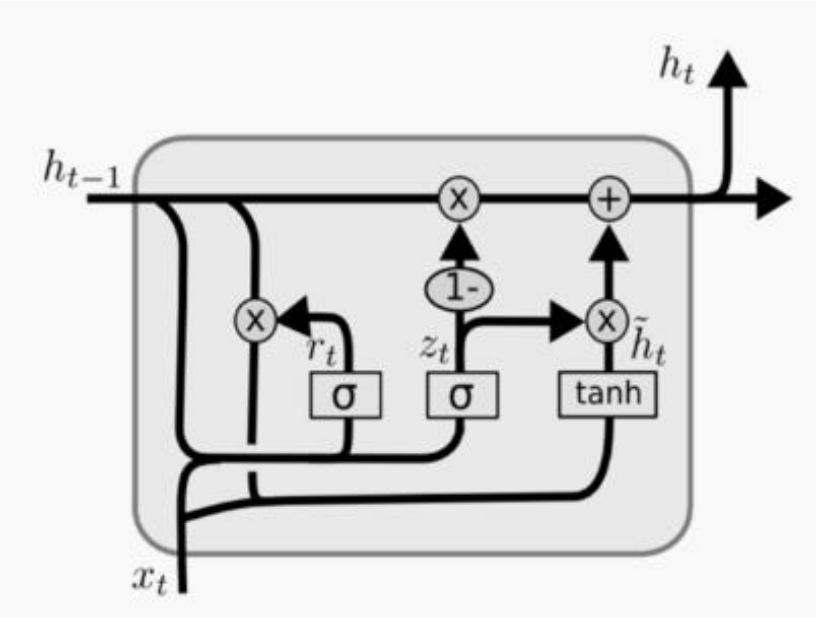


Figure 21 : Architecture d'une cellule GRU **(Hairy, 2021)**

4.7.3.3. Applications des RNN dans le traitement du langage naturel

Les réseaux de neurones récurrents (RNN) sont largement utilisés dans le domaine du traitement du langage naturel (NLP) en raison de leur capacité à traiter des données séquentielles et contextuelles (Mathur, 2022).

- Traduction automatique
- Création de texte
- Légende des images
- Reconnaissance de la parole
- Prévision des séries chronologiques

4.8. Les transformers

4.8.1. Définition et architecture des transformers

Les réseaux de neurones séquentiels, ainsi que les CNN pour le texte, ont été pendant quelques temps la meilleure alternative pour le NLP. Puis en 2017, Vaswani et ses collègues ont publié un article de référence « *Attention is all you need* » (Noam , Parmar, & Vaswani, 2017), sur lequel s'appuient aujourd'hui les modèles de NLP. En effet, ce fut le début du bloc Transformer, et des modèles utilisant l'auto-attention. Le Transformer est constitué d'une partie permettant l'encodage des données, composée d'encodeurs, et d'une seconde partie dédiée au décodage, composée de décodeurs Le schéma du Transformer est visible en figure 22.

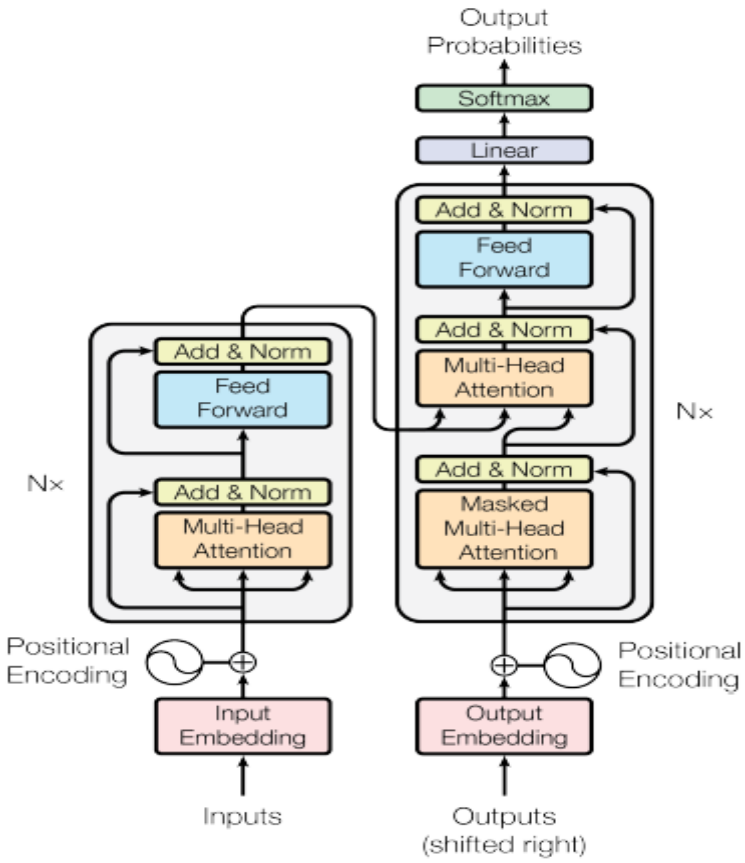


Figure 22 Architecture de transformer **(Noam , Parmar, & Vaswani, 2017)**

- ***L'encodeur***

Chaque encodeur est composé de deux blocs. Tout d'abord, un bloc d'auto-attention, élément majeur du bloc Transformer, suivi d'un bloc de réseau à propagation avant (*feed-forward neural network*). Si le principe de l'attention consiste à mesurer le lien entre deux éléments de deux séquences, l'auto-attention correspond au même mécanisme appliqué à une seule séquence. En effet, l'auto-attention permet de déterminer les relations entre les différents *tokens* d'une même séquence. Ici, l'auto-attention est *multi-head,* cela signifie que les calculs sont effectués en parallèle par plusieurs têtes d'attention.

- ***Le décodeur***

Pour le décodeur, il contient également un bloc d'auto-attention et un bloc de réseau à propagation avant, avec en plus un bloc d'attention Encoder-Decoder permettant de faire le lien entre la séquence encodée en entrée, et la séquence de sortie qui est décodée.

4.8.2. Avantages des Transformers par rapport aux RNN

Avant l'avènement du Transformer, les modèles de NLP se basaient principalement sur les RNN tels que le LSTM et le GRU. Cependant, ces modèles présentaient une limitation majeure : ils étaient incapables de gérer efficacement des séquences d'entrée de longueurs variables. Les RNN traitent les séquences élément par élément, ce qui limité leur capacité à traiter des séquences plus longues.

Le Transformer a révolutionné cette approche en introduisant le mécanisme d'attention, utilisé à la fois dans l'encodeur et le décodeur. Ce mécanisme permet de calculer le poids d'attention pour chaque élément de la séquence d'entrée, ce qui permet de déterminer l'importance relative de chaque partie de la séquence.

Le modèle Transformer offre ainsi deux avantages majeurs par rapport aux modèles RNN traditionnels (Les « Transformers » : pourquoi l'Attention □ est tout ce dont vous avez besoin?, 2023):

- Il peut traiter des séquences d'entrée de longueurs variables de manière efficace, sans être limité par la longueur de la séquence.
- Grâce au mécanisme d'attention, il est capable de mettre l'accent sur les parties les plus pertinentes de la séquence d'entrée, ainsi sa capacité à capturer les relations et les dépendances entre les mots.

5. BERT

BERT est un modèle avancé d'intégration de mots basé sur l'architecture codée du transformateur. Nous utilisons BERT comme encodeur de phrase, qui peut obtenir avec précision la représentation contextuelle d'une phrase. BERT supprime la contrainte unidirectionnelle en utilisant un modèle de langage de masque (MLM). Il masque aléatoirement certains jetons de l'entrée et prédit l'identifiant de vocabulaire original du mot masqué uniquement. MLM a augmenté la capacité de BERT à surperformer par rapport aux méthodes d'intégration précédentes. C'est un système profondément bidirectionnel qui est capable de gérer le texte non étiqueté en conditionnant conjointement le contexte gauche et droit dans toutes les couches (R, M, & Z, 2021).

6. Conclusion

En conclusion, l'arrivée de l'apprentissage automatique et de l'apprentissage profond a révolutionné le domaine du traitement du langage naturel. Deux approches clés, les réseaux de neurones récurrents (RNN) et le modèle Transformer, ont joué un rôle crucial dans cette évolution. Les RNN ont été largement utilisés dans les premiers modèles de NLP en raison de leur capacité à traiter des séquences temporelles. Ils sont adaptés pour capturer les dépendances séquentielles à travers le temps. Cependant, les RNN souffrent de limitations, notamment leur difficulté à gérer efficacement des séquences de longueurs variables en raison de la propagation de l'information à travers les étapes temporelles.

L'introduction du modèle Transformer a dépassé ces limitations en utilisant le mécanisme d'attention. Contrairement aux RNN, le Transformer traite les séquences globalement plutôt que séquentiellement, permettant ainsi une parallélisassions efficace. Le mécanisme d'attention permet au Transformer de se concentrer sur les parties les plus importantes de la séquence d'entrée, capturant ainsi les relations complexes entre les mots et diminuant les performances du modèle sur les différentes tâches de NLP.

L'apprentissage automatique, en particulier l'apprentissage profond, une ouverture de

nouvelles perspectives et possibilités dans le traitement du langage naturel. Grâce aux progrès continus dans ces domaines, nous avons vu des améliorations significatives dans des applications telles que la traduction automatique, la génération de texte, l'analyse de sentiments, la reconnaissance vocale et bien d'autres.

Chapitre 03

Contribution et Expérimentation

1. Introduction

Dans le cadre de notre étude, nous concentrons sur l'analyse automatique des sentiments dans les tweets arabe en tenant compte des émojis. Avec l'arrivée des réseaux sociaux, les utilisateurs du monde entier ont adopté Twitter comme une plateforme pour partager leurs opinions et sentiments. Cependant, l'analyse des sentiments dans les tweets arabes présente des défis particuliers en raison des spécificités linguistiques propres à cette langue et de l'utilisation fréquente d'émojis pour exprimer les émotions.

L'objectif de notre étude est d'entraîné et évalué un modèle d'apprentissage profond capable de comprendre et d'interpréter les sentiments exprimés dans les tweets arabe, en intégrant efficacement les informations contenues dans les émojis. Notre modèle est basée sur la méthode de transformer.

2. Architecture du système

2.1. Schéma du système

c

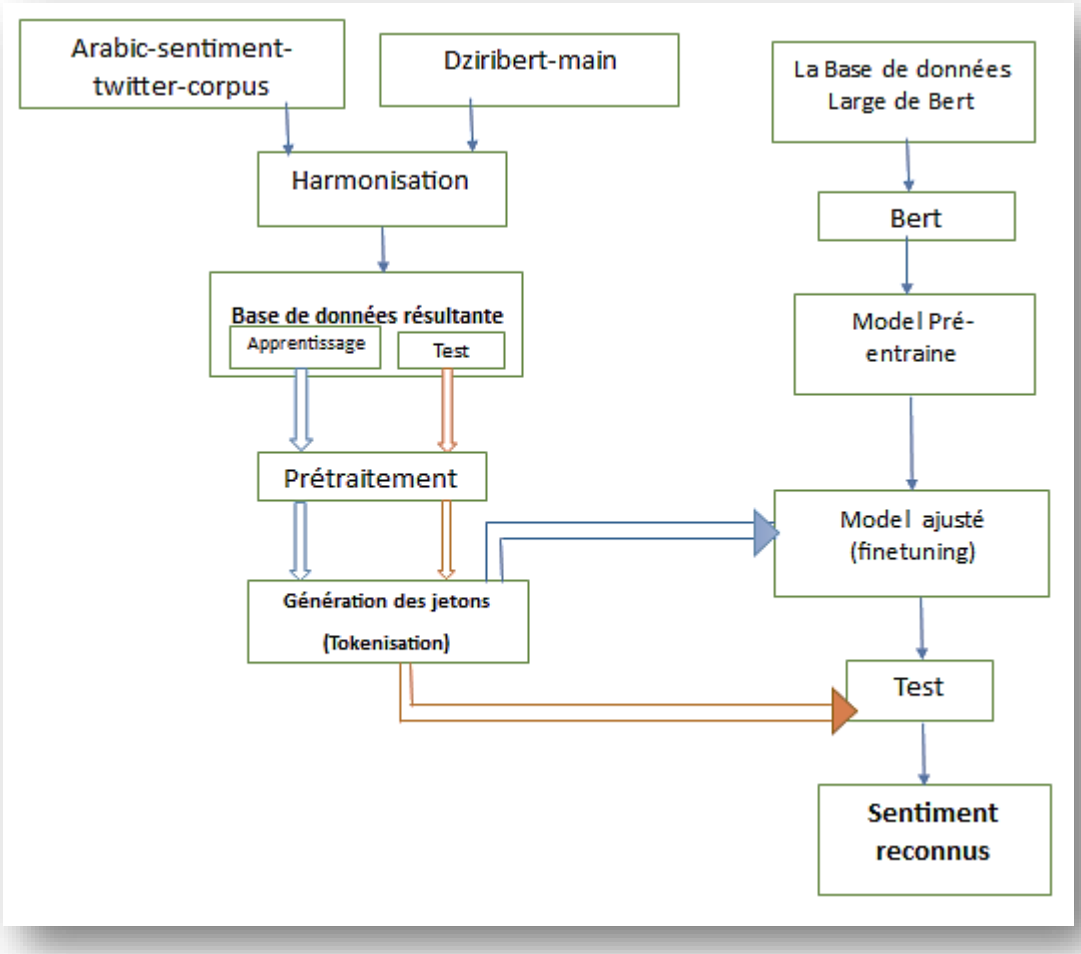


Figure 23: Architecture générale du système

2.2. Base des données

Pour réaliser cette étude, nous avons construit un corpus de tweets arabes en utilisant Arabic-sentiment-twitter-corpus (HANY, Arabic Sentiment Analysis using Arabic-BERT) , enrichi avec des commentaires de la base de DziriBert. La base de donne finale est annotée en termes de polarité sentimentale, en prenant en considération à la fois du texte et des émojis y intégrés.

Le tableau suivant résume les bases de données utilisées dans cette étude :

Base de données	Contenu	Type de fichier	Nombre de lignes	Nombre de colonnes	Nombre des tweets positifs	Nombre des tweets négatifs
Arabic-sentiment-twitter-corpus (HANY, Arabic Sentiment Analysis using Arabic-BERT)	- Train_pos - Train_neg - Test_pos - Test_neg	Tsv	58751	2	29849	28902
Dziribert-main (Abdaoui)	- Train_sent - Test_sent	Csv	6965	2	4350	2615

Tableau 2: Les informations des bases de données

2.2.1. Exploration des données

Dans cette section nous allons découvrir, visualiser et tirer des informations significatives à partir de notre corpus, afin de réaliser des modifications appropriées et prendre des décisions éclairées. Les figures 24 et 25 représentent des exemples sur les deux bases utilisées.

	0	1
0	neg	😭 اعترف ان بتس كانو شوي شوي يجيبو راسي لكن اليوم بالزايد
1	neg	💔 #Avlu توقعت اذا جات داريا بشوفهم كاملين بس لي للحين احس فيه احد ناقصهم
2	neg	الاهلي_الهلال اكتب توقعك لنتيجة لقاء الهلال والاهلي تحت التاق 👇 #تحدي_اسرع_روقان وادخل في سحب# على X قيمة ايفون...
3	neg	نعمة المضادات الحيوية . تضع قطرة 💧 مضاد بنسلين على بكتيريا 🦠 فتنفجر 💥 و تموت . الأخيرة يبدو انها بكتيريا مقاومة فأخذ...
4	neg	💔 الدودو جايه تكمل علي

Figure 24 : des exemples de la base Arabic-sentiment-twitter-corpus

	Unnamed: 0	text	label
0	0	@samraaroshdy @A7MD__MOkhtar انا تعساء و طالت تعاستنا بسبب ما تعانه الامة من تقتيل و تشريد و تدمير . اللهم اعنا عل... https://t.co/OQFkqDoUsQ	0
1	1	sur commande https://t.co/HXLeowAkly	2
2	2	#zara #Barcelona https://t.co/R4b2ECfZjW	1
3	3	@Ania27El حلقة من مسلسل طويل و ممل !	0
4	4	https://t.co/spOZ4VE0rm	1

Figure 25 : des exemples de la base Dziribert-main

2.2.1.1. Harmonisation

L'application d'une séquence de modifications a produit une nouvelle Base de données homogène.

- Modifications appliquées sur Arabic-sentiment-twitter-corpus :
- Changement des noms des colonnes «0» et «1» par «label» et «tweet» respectivement.
- Remplacement du contenu de la colonne « label» de «neg» et «pos » à «0» et «1» respectivement.

	label	tweet
0	0	😭 اعترف ان بتس كانو شوي شوي يجيبو راسي لكن اليوم بالزايد
1	0	💔 #Avlu توقعت اذا جات داريا بشوفهم كاملين بس لي للحين احس فيه احد ناقصهم
2	0	الاهلي_الهلال اكتب توقعك لنتيجة لقاء الهلال والاهلي تحت التاق 👇 #تحدي_اسرع_روقان وادخل في# على X سحب قيمة ايفون...
3	0	نعمة المضادات الحيوية . تضع قطرة 💧 مضاد بنسلين على بكتيريا 🦠 فتنفجر 💥 و تموت . الأخيرة يبدو انها بكتيريا مقاومة فأخذ...
4	0	💔 الدودو جايه تكمل علي

Figure 26: des exemples de la base Arabic-sentiment-twitter-corpus après les modifications

- Modifications appliquées sur Dziribert-main
- Suppression de colonne nommé « Unnamed: 0 ».
- L'inversement des colonnes « label» et « text ».
- Changement du nom de la colonne « text » par « tweet ».
- Suppression des tweets neutres « label=1 ».
- Remplacement des valeurs « label = 2 » par « label = 1 ».

	label	tweet
0	0	@samraaroshdy @A7MD__MOkhtar انذا تعساء و طالت تعاستنا بسبب ما تعانه الامة من تقتيل و تشريد و تدمير . اللهم اعنا عل... https://t.co/OQFkqDoUsQ
1	1	sur commande https://t.co/HXLeowAkly
3	0	@Ania27El حلقة من مسلسل طويل و ممل !
5	1	@aljoumhouria @elissakh الله معاك اليسا حبيبتي كوني قوية لا تحزني
7	1	Let's kick off the nuts-eating habit 🥜 https://t.co/ZSJHtdyMlh

Figure 27 : des exemples de la base Dziribert-main après les modifications

Après ces modifications on a concaténé tous les fichiers obtenue et les stockées dans une base de données nommé « df_finale » et l'histogramme dans la figure 29 représente la distribution des tweets en positifs et négatifs.

	label	tweet
0	0	@ainourelsallak كونتينيو ليس كميسي
1	1	@LotfyMwafy اه صح ... اكيد
2	1	تبتسم ولك حسنة .. وتمرض ولك أجر .. وتصبر ولك من بعد العسر يسر .. أيُّ دين أجمل من هذا ! " ..😉😍 " صباحكم .. وكل ص... https://t.co/zQzUdE3hZu
3	1	@yKeqyIheog2W2lc انت كل عام ونت بالف خيييير
4	1	اللهم اكتب لنا من أنفاس هذا الصباح خيراً نعلمه و رزقاً نكتسبه ،و بشارة نسعد بها. صباح الخير ع الجميع

Figure 28 : Base de données finale

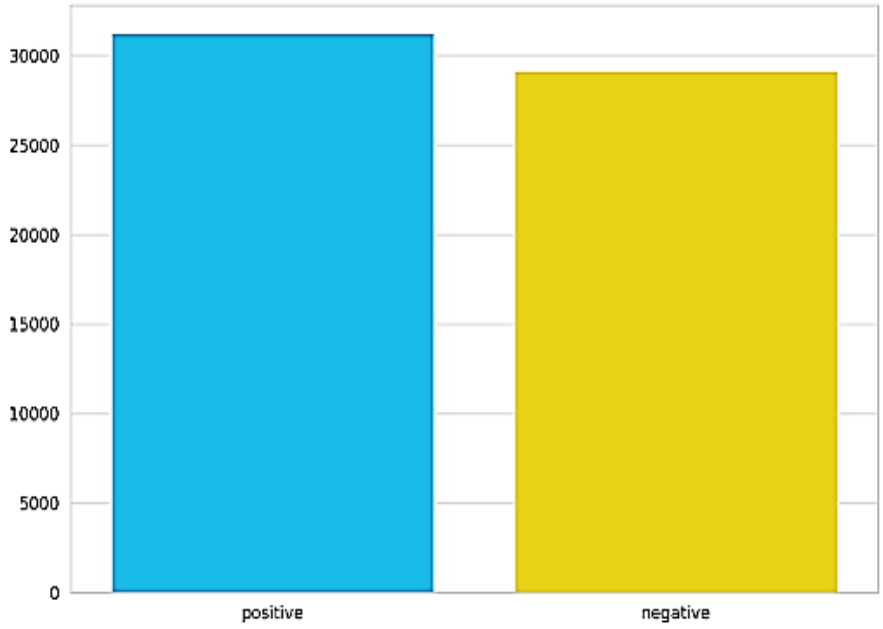


Figure 29 : La distribution des tweets

2.2.1.2. Nuage de tags

Le nuage de tags est une représentation visuelle généralement présenté sous forme de diagramme où les mots clés apparaissent dans différentes tailles ou couleurs en fonction de leur fréquence ou de leur importance. Voici un exemple de ce nuage :

Figure 30 : exemple du nuage de tags

2.2.1.3. Répartissions de la base de données

La base de données finale sera divisée en trois sous-bases pour l'apprentissage, la validation et le test nommées respectivement « train_df», «valid_df » et «test_df». La figure ci-dessous montre le pourcentage et le nombre de tweets dans chaque sous-base.

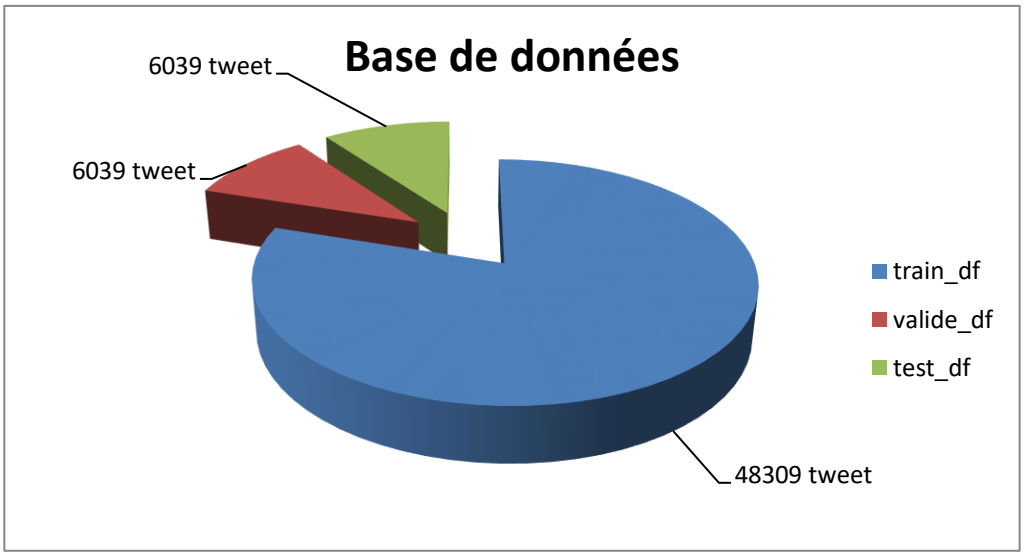


Figure 31 : Division de base de données

2.3. Prétraitement

Le prétraitement est une étape importante dans l'analyse du sentiment, et pour cela nous avons développé une fonction de prétraitement qui comprend plusieurs étapes spécifiques axées sur la sémantique des tweets. La figure ci-dessous représente le processus de prétraitement.

Figure 32 : Processus de prétraitement

• **Supprimer les mentions d'entités (noms d'utilisateur) :**

Étant donné notre intérêt pour le sentiment exprimé dans le tweet lui-même plutôt que par l'utilisateur, nous avons pris la décision de supprimer l'identifiant utilisateur, qui se présente sous la forme d'un nom précédé d'une arobase "@" et qui est un lien direct vers un compte Twitter. Par exemple :

Phrase originale	" حلقة من مسلسل طويل و ممل @Anla27El"
Phrase après la modification	" حلقة من مسلسل طويل و ممل"

Tableau 3: exemple de suppression des mentions d'entités

- **supprimer les lettres répétées**.

La répétition des lettres dans un tweet est sans importance et n'apporte rien au sentiment exprimé. Au contraire, elle ne fait qu'augmenter le volume des données. Par conséquent, nous procédons à l'élimination de tous les termes répétés dans chaque tweet. Par exemple :

Phrase originale	"الله على صوتهاااا"
Phrase apres la modification	"الله على صوتها"

Tableau 4: exemple de suppression des lettres répétées

- **Supprimer les mots avec hashtags**

L'expression du sentiment par un utilisateur est influencée par les mots qu'il utilise, mais il arrive que certains éléments dans un tweet n'aient aucune valeur sentimentale, tels que les mots comportant des hashtags "#". Par exemple :

Phrase originale	"يخبرني_الصباح# تأخر الاحلام يجعلنا نكرهها"

Phrase apres la modification	" تأخر الاحلام يجعلنا نكرهها"

Tableau 5 : exemple de Suppression des mots avec hashtags

• **Supprimer les URL**

Lors de l'analyse des sentiments dans les médias sociaux, les liens partagés ne sont pas nécessaires pour l'opinion exprimée et elles peuvent être effectuées comme du bruit.

Phrase originale	"تبتسم و لك حسنة و تمرض و لك اجر و تصبر و مك من بعد العسر يسر أي دين اجمل من هدا https://t.co/zQzUdE3hZu "
Phrase après la modification	"تبتسم و لك حسنة و تمرض و لك اجر و تصبر و لك من بعد العسر يسر أي دين اجمل من هدا"

Tableau 6: exemple de Suppression les URL

• **Supprimer la ponctuation et les caractères non alphabétiques sans supprimer les émojis.**

Les caractères non alphabétiques tels que les chiffres, la ponctuation et les caractères spéciaux n'apportent aucune contribution à l'expression d'un sentiment. Par conséquent, ils sont insignifiants dans ce contexte. Par exemple :

Phrase originale	"الاتحاد يتوج بطلا للدوري -4/12 ☺☺"
Phrase apres la modification	" الاتحاد يتوج بطلا للدوري☺☺"

Tableau 7 : exemple de Suppression des caractères non alphabétiques

2.4. Génération des jetons (Toknisation)

La Toknisation est une étape essentielle du prétraitement des données textuelles, et elle

permet de transformer un texte continu en une séquence discrète de jetons (tokens), facilitant ainsi leur traitement. Les jetons spéciaux utilisés sont illustrées dans le tableau ci-dessous.

Le jeton	Id de jeton	Signification
CLS	2	Pour le début de la phrase
SEP	3	Pour la fin de la phrase
MASK	4	Masquer certaines parties dans le texte lors de l'entrainement pour effectuer des taches tel que la prédiction des mots manquants
PAD	0	Pour le rembourrage
UNK	1	Pour les mots inconnus

Tableau 8 : les jetons spéciaux utilisés

Voici un exemple de toknisation :

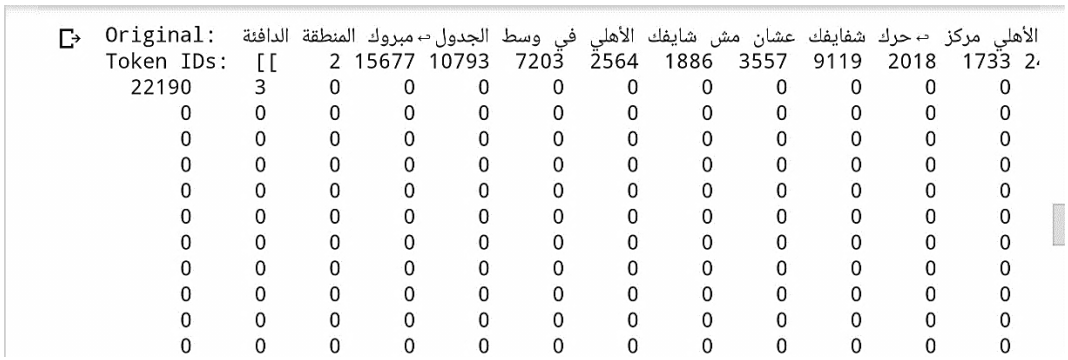


Figure 33: Résultat de Toknisation

2.5. Classification

2.5.1. Apprentissage

La phase d'apprentissage consiste à entraîner un modèle sur un ensemble de données d'entraînement afin qu'il puisse apprendre des schémas et des relations. Ensuite, la phase de validation vise à évaluer les performances du modèle sur un ensemble de données distinctes, mais similaires, afin de déterminer son aptitude à généraliser et à répondre de manière fiable à de nouvelles données. Cette phase regroupe plusieurs étapes :

a) Création des DataLoaders

Cette étape concerne la préparation des données pour l'entraînement et la validation d'un

modèle utilisant BERT (Bidirectional Encoder Representations from Transformers). Voici une explication en points :

- Conversion des autres types de données en torche.Tensor :

Les étiquettes d'entraînement (train_labels) et de validation (valide_labels) sont converties en tenseurs torch.Tensor à l'aide de la fonction torch.tensor().

- Définition de la taille du lot (batch_size) :

Pour l'affinage (fine-tuning) de BERT, il est recommandé d'utiliser une taille de lot (batch size) de 16 ou 32. Cela signifie que les données seront divisées en lots de 16 ou 32 échantillons lors de l'entraînement et de la validation.

- Création du DataLoader pour l'ensemble d'entraînement :

Les données d'entraînement sont regroupées en utilisant la classe TensorDataset, qui prend en paramètre les tenseurs train_inputs, train_masks et train_labels.

Un échantillon aléatoire (RandomSampler) est utilisé pour associer les données d'entraînement avant chaque époque et garantir un apprentissage robuste.

Le DataLoader (train_dataloader) est créé en utilisant le TensorDataset, l'échantillonneur et la taille de lot souhaité. Il est responsable de l'itération sur les données d'entraînement par lots lors de l'entraînement du modèle.

- Création du DataLoader pour l'ensemble de validation :

Les mêmes étapes sont répétées pour l'ensemble de validation.

Les données de validation sont regroupées en utilisant TensorDataset avec les tenseurs valide_inputs, valide_masks et valide_labels.

Un échantillon séquentiel (SequentialSampler) est utilisé pour parcourir les données de validation dans l'ordre.

Le DataLoader (valide_dataloader) est créé avec le TensorDataset, l'échantillon séquentiel et la taille de lot sélectionnée. Il est utilisé pour itérer sur les données de validation par lots pendant l'évaluation du modèle.

En résumé, cette partie du code permet de convertir les données dans un format compatible avec BERT, de les regrouper en ensembles d'entraînement et de validation, et de créer des objets DataLoader pour itérer sur ces ensembles par lots pendant l'entraînement et la validation du modèle.

- Choix et création du modèle

Dans cette étape nous définirons la classe et les fonctions d'initialisation du modèle en créant une structure de modèle. Dans le contexte de l'apprentissage automatique, cette étape est cruciale pour définir l'architecture et les paramètres du modèle que l'on souhaite entraîner.

La classe du modèle est une entité qui encapsule toutes les fonctionnalités et les opérations

du modèle. Elle peut inclure des couches de neurones, des opérations mathématiques, des paramètres ajustables, des fonctions d'activation, etc.

b) Initialisation et optimisation du modèle

Cette étape présente la fonction utilisée pour initialiser les composants principaux nécessaires à l'entraînement du modèle BERT Classifier, tels que le modèle lui-même, l'optimiseur et le scheduler du taux d'apprentissage Voici les points principaux du code :

- Instanciation du modèle BertClassifier en utilisant la classe préalablement définie.
- Spécification de l'utilisation du GPU pour réaliser le modèle.
- Création de l'optimiseur AdamW qui optimise les paramètres du modèle.
- Définition du taux d'apprentissage et de la valeur epsilon pour l'optimiseur.
- Calcul du nombre total d'étapes d'entraînement en multipliant le nombre de lots (train_dataloader) par le nombre d'époques.
- Configuration du scheduler du taux d'apprentissage en utilisant la fonction "get_linear_schedule_with_warmup". Ce scheduler ajuste le taux d'apprentissage linéairement pendant les étapes d'entraînement, avec une phase de "warmup" initial où le taux d'apprentissage augmente progressivement.
- Retour du modèle initialisé, de l'optimiseur et du scheduler.

c) Finetuning

Dans cette phase on effectue l'entraînement itératif du notre modèle BERT en utilisant des dataloaders pour fournir les données d'entraînement et de validation. Il calcule les pertes et les précisions à chaque époque et affiche les résultats pour suivre la progression de l'entraînement. Voici les rtapes principales de cette phase :

- Initialisation des listes pour stocker les pertes et des précisions d'entraînement et de validation.
- Début de la boucle d'entraînement sur le nombre spécifié d'époques.
- Pour chaque époque : on commence par l'affichage de l'en-tête du tableau des résultats et la réinitialisation des variables de suivi au début de chaque époque, ensuite on passe en mode entraînement du modèle et pour chaque lot de données d'entraînement on applique les opérations suivantes :
  - Chargement du lot sur le GPU.
  - Réinitialisation des gradients calculés précédemment.
  - Exécution de la propagation avant pour obtenir les logits.
  - Calcul de la perte et accumulation des valeurs de perte.
  - Exécution de la rétropropagation pour calculer les gradients.
  - Mise à jour des paramètres et du taux d'apprentissage.
  - Calcul de la précision d'entraînement.
  - Affichage des valeurs de perte et du temps délivrées toutes les 20 itérations.

  Après on Calcul la perte moyenne sur l'ensemble des données d'entraînement, et Stocker les valeurs de perte et de précision d'entraînement et on voire si l'évaluation est activée on passe à les deux étapes suivants :

- Appel de la fonction "évaluer" pour mesurer la performance du modèle sur l'ensemble de validation.
- Affichage de la performance sur l'ensemble des données d'entraînement et de validation.

- Après la fin de la boucle d'entraînement, affichage du message "Entraînement terminé !".
- passe à la partie de validation et définir la fonction "évaluer" qui mesure la performance du modèle sur l'ensemble de validation : au départ en passer le modèle en mode évaluation et pour chaque lot dans l'ensemble de validation on applique les opérations suivantes :

- Chargement du lot sur le GPU.
- Calcul des logits sans effectuer de gradients.
- Calcul de la perte.
- Obtention des prédictions.
- Calcul du taux de précision.

Après on calcul la précision et la perte moyenne sur l'ensemble de validation, enfin on retour les valeurs de perte et de précision.

2.5.2. Evaluation

Ces étapes permettent de charger le modèle BERT pré-entraîné, d'évaluer ses performances sur les ensembles de validation et de test, de prédire les sentiments des tweets et de visualiser la matrice de confusion.

- *Chargement du modèle BERT pré-entraîné :* La première étape consiste à charger un modèle BERT pré-entraîné, qui servira à la classification des sentiments des tweets.
- *Prédiction des probabilités de sentiment pour l'ensemble de validation* : En utilisant le modèle BERT pré-entraîné, une passe avant (forward pass) est effectuée sur l'ensemble de validation. Cela permet d'obtenir les logits, c'est-à-dire les valeurs brutes de sortie, pour chaque lot de données de validation. Les logits sont ensuite convertis en probabilités en appliquant la fonction softmax.
- *Évaluation des performances du modèle sur l'ensemble de* validation :La fonction `evaluate_roc` est utilisée pour calculer l'Aire Sous la Courbe (AUC) et l'exactitude (accuracy) en utilisant les probabilités prédites et les étiquettes réelles de l'ensemble de validation. Elle trace également la courbe ROC en utilisant les taux de faux positifs et les taux de vrais positifs.
- *Prétraitement des données de test (tokenization)* : Les données de test sont prétraitées en effectuant la tokenization à l'aide de BERT. Cela consiste à convertir les phrases en séquences de tokens et à générer des masques d'attention correspondants.
- *Création d'un DataLoader pour l'ensemble de test* : Un DataLoader est créé à partir des entrées tokenisées et des masques d'attention des données de test. Cela permet de préparer les données pour l'inférence avec le modèle BERT.
- *Prédiction des probabilités de sentiment pour l'ensemble de test :* En utilisant le modèle BERT pré-entraîné, une passe avant est effectuée sur l'ensemble de test à l'aide du DataLoader. Les probabilités prédites pour chaque tweet sont obtenues à partir des logits générés par le modèle.

- *Conversion des probabilités en prédictions de sentiment* : Les probabilités prédites sont converties en prédictions de sentiment en fonction d'un seuil de probabilité prédéfini. Par exemple, si la probabilité est supérieure à 0,5, la prédiction sera "positive", sinon elle sera "negative".
- *Calcul du ratio de tweets prédits comme non négatifs* : Le ratio des tweets prédits comme non négatifs est calculé en divisant la somme des prédictions positives par le nombre total de tweets. Cela permet de quantifier la proportion de tweets classés comme ayant un sentiment positif.
- *Évaluation des performances du modèle sur l'ensemble de test* : La fonction `evaluate_roc` est utilisée pour évaluer les performances du modèle BERT sur l'ensemble de test. Elle calcule l'AUC et l'exactitude en comparant les probabilités prédites avec les étiquettes réelles des données de test. La courbe ROC est également tracée pour visualiser la performance du modèle.
- *Visualisation de la courbe ROC :* La courbe ROC est tracée en utilisant les taux de faux positifs et les taux de vrais positifs calculés lors de l'évaluation du modèle sur l'ensemble de test. Cela permet de visualiser graphiquement les performances du modèle en termes de discrimination entre les classes positives et négatives.
- *Prédiction du sentiment pour un tweet donné :* La fonction `predict_tweet_sentiment` prend un tweet en entrée et effectue les étapes nécessaires pour prédire son sentiment en utilisant le modèle BERT pré-entraîné. La prédiction de sentiment ("positive" ou "negative") est renvoyée comme résultat.
- *Affichage d'une matrice de confusion :* La fonction `show_confusion_matrix` prend une matrice de confusion en entrée et la visualise en utilisant une heatmap (carte thermique) grâce à la bibliothèque seaborn. Cela permet de visualiser la répartition des prédictions par rapport aux étiquettes réelles et d'analyser les performances du modèle en termes de classifications correctes et incorrectes.

3. expérimentations et résultats

Les modèles ont été réalisée en utilisant des paramètres spécifiques, notamment un dropout, un learning rate, nombres des époques et une taille de batch. Les résultats obtenus ont été analysés en termes de perte d'entraînement, perte de validation, exactitude de validation et exactitude d'entraînement.

3.1. Définition des hyperparamètres

- *La taille de lot (batch size)*

Fait référence au nombre d'échantillons d'entraînement qui sont propagés à travers le modèle en une seule itération. Il détermine la quantité de données traitées en parallèle lors de l'entraînement du modèle.

- *Le pas d'apprentissage (learning rate)*

Est un hyperparamètre qui détermine la vitesse à laquelle un modèle d'apprentissage automatique met à jour ses poids lors de l'entraînement. Le pas d'apprentissage contrôle l'amplitude des modifications apportées aux poids du modèle à chaque itération, ce qui peut affecter la convergence et la performance globale du modèle.

• *Le dropout*

Est une technique de régularisation utilisée dans les réseaux de neurones pour éviter la sur adaptation. Elle consiste à aléatoirement désactiver un pourcentage de neurones pendant l'entraînement, ce qui encourage le réseau à apprendre des caractéristiques plus robustes et à généraliser mieux. Cela contribue à améliorer les performances du modèle en évitant une dépendance excessive à certains neurones spécifiques et en favorisant une meilleure généralisation.

• *Les époques*

Le numéro d'époque représente le nombre de fois que l'ensemble des données d'entraînement est parcouru pendant l'apprentissage d'un modèle. Il est utilisé pour suivre le progrès de l'entraînement, ajuster les poids du modèle et effectuer des évaluations périodiques pour mesurer les performances.

3.2. Ajustement des hyperparamètres

• Tout d'abord, nous avons fixé le nombre d'époques, le taux d'apprentissage et la vitesse d'apprentissage, puis nous avons modifié la taille du lot.

Epocs	Dropout	Learning rate	Batch size	Loss train	Loss validation	Accuracy validation	Accuracy train
3	0,1	2 e-5	32	0,440033	0, 475580	76,63	78,970621
			16	0,411205	0,468919	77,33	80,568671

Tableau 9 : Ajustement de La taille de lot.

• Après avoir fixé la taille du lot (batch size), nous avons modifié la valeur du pas d'apprentissage (learning rate) comme suit :

Epocs	Dropout	Batch size	Learning rate	Losss train	Loss validation	Accuracy validation	Accuracy train
			2 e-5	0,411205	0,468919	77,33	80,568671

3	0,1	16	3 e-5	0,371414	0,478568	77,18	82,967473
			5 e-5	0,325579	0,497562	78,77	85,4889863

Tableau 10 : Ajustement du pas d'apprentissage

Nous constatons que les meilleurs résultats sont obtenus lorsque nous avons fixé le pas d'apprentissage à sa valeur maximale (learning rate = 5e-5). Cependant, nous sommes restés sur cette valeur fixe.

- Une fois que nous avons déterminé le pas d'apprentissage (learning rate), nous avons ajusté la valeur de dropout de la manière suivante :

Batch size	Learning rate	dropout	époques	Losss train	Loss validation	Accuracy validation	Accuracy

Epoques	Batch size	Learning rate	Dropout	Losss train	Loss validation	Accuracy validation	Accuracy train
3	16	5 e-5	0,1	0,325579	0,497562	78,77	85,4889863
			0,15	0.324675	0.506631	78.75	85.168848
			0,2	0.326423	0.499358	78.93	78.93
			0,25	0.322203	0.498855	78.67	85.531257
			0,3	0.326801	0.490318	79.20	85.369036

Tableau 11: Le dropout

En examinant le tableau, nous constatons que l'exactitude la plus élevée (79%) est obtenue lorsque le dropout est fixé à 0,3. Par conséquent, nous décidons de le maintenir à cette valeur pour obtenir les meilleurs résultats en termes de précision.

Après avoir sélectionné le dropout nous avons apporté des modifications à la valeur nombre d'époques comme suit :

							train
16	5 e - 5	0,3	3	0.326801	0.490318	79.20	85.369036
			6	0.193507	0.709212	79.07	92.067042
			10	0.105538	1.210668	78.72	96.616137

Tableau 12 : Ajustement de nombre d'époques

3.3. Choix du modèle

Après cette analyse, on conclut que les paramètres optimaux pour un modèle performant sont les suivants :

Epocs	Batch size	Learning rate	Dropout	Losss train	Loss validation	Accuracy validation	Accuracy train	AUC_ROC
3	16	5 e-5	0.3	0.326801	0.490318	79.20	85.369036	0.8727

Tableau 13 : hyper paramétrés pour le modèle choisi.

3.4. La courbe ROC pour le modèle choisi

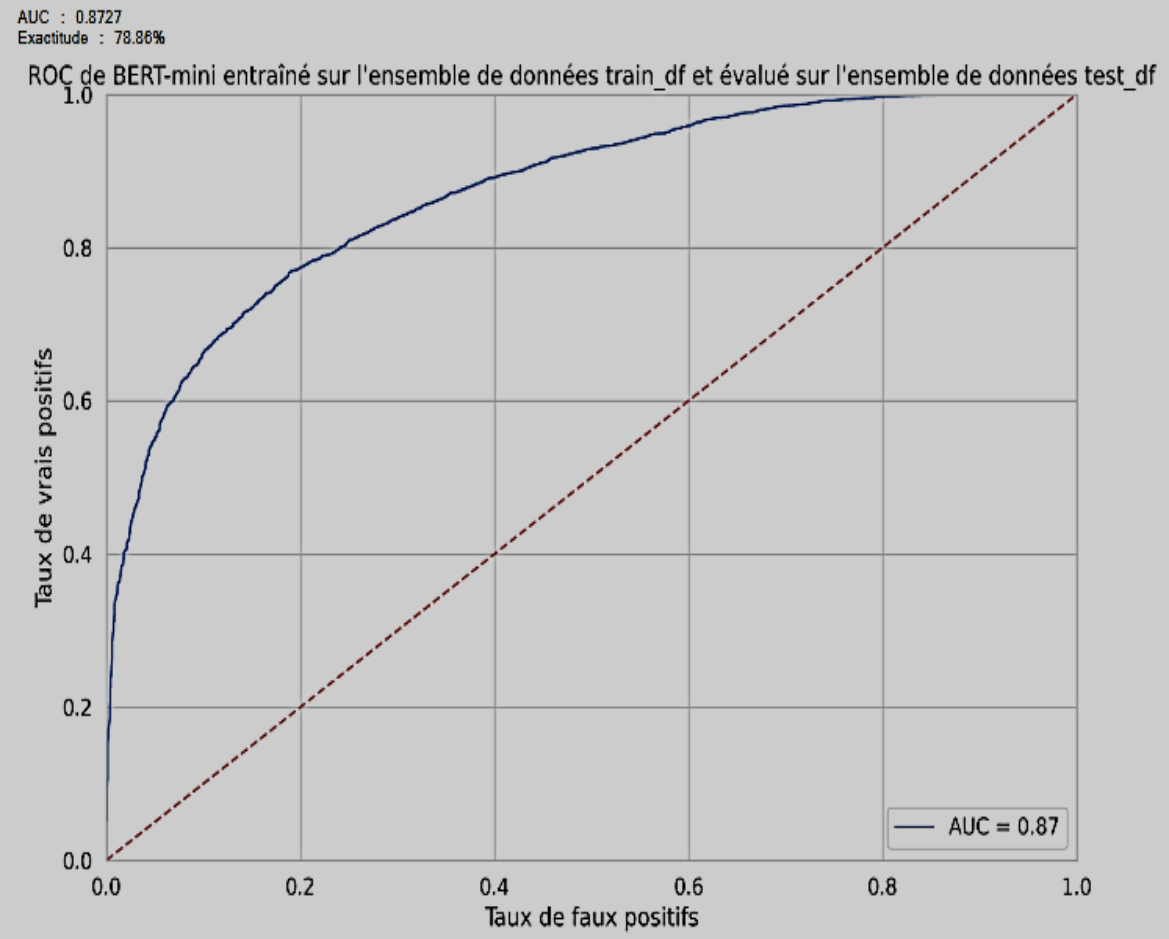


Figure 34 : Le courbe ROC pour l'ensemble d'apprentissage et de test avec les emojis

La courbe ROC est un outil couramment utilisé dans l'analyse des performances des modèles de classification Elle permet d'évaluer la capacité d'un modèle à discriminer entre différentes classes et représente graphiquement la relation entre le taux de vrais positifs (TVP) et le taux de faux positifs (TFP) pour différents seuils de classification. Cette courbe est tracée en représentant le TFP sur l'axe des abscisses et le TVP sur l'axe des ordonnées. Chaque point sur la courbe correspond à un seuil de classification différent. En général, plus la courbe ROC se rapproche du coin supérieur gauche du graphique, meilleure est la performance du modèle, car cela signifie que le modèle a un TVP élevé tout en maintenant un faible TFP.

Une mesure couramment utilisée pour résumer la performance d'un modèle à partir de la courbe ROC est l'AUC-ROC (Area Under the Curve of Receiver Operating Characteristic). L'AUC-ROC représente l'aire sous la courbe ROC et fournit un score qui mesure la capacité globale du modèle à discriminer entre les classes. Un score AUC-ROC de 1 indique une performance parfaite, tandis qu'un score de 0,5 indique une performance équivalente à une prédiction aléatoire.

3.5. La matrice de confusion pour le modèle choisi

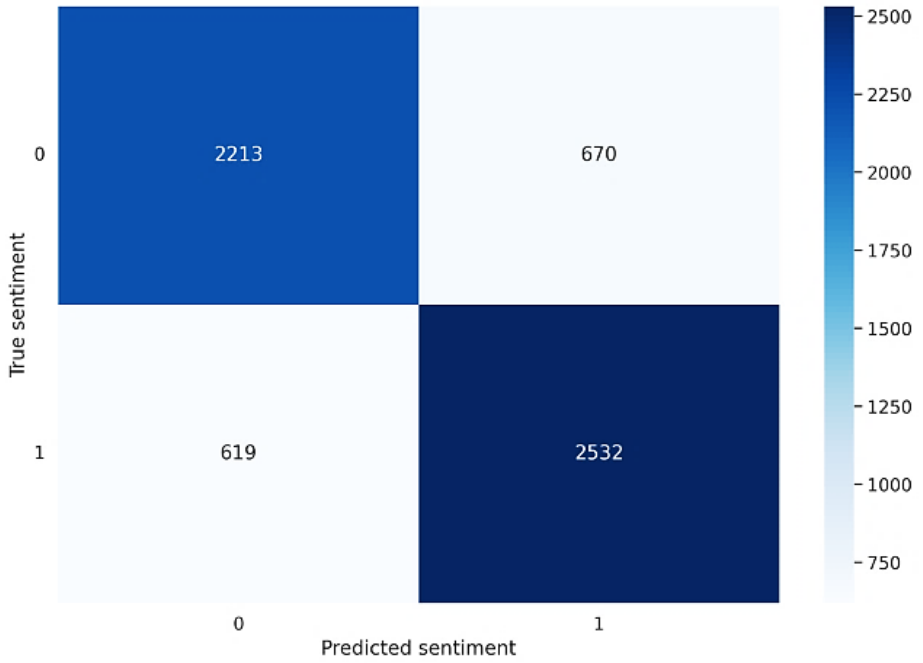


Figure 35 : Matrice de confusion pour modèle choisi avec les émojis

Cette matrice représente des informations détaillées sur les performances du modèle, telles que les tweets classés à tort comme positifs (FP), les tweets classés à tort comme négatifs (FN), les tweets correctement classés comme positifs (VP) et les tweets correctement classés comme négatifs (VN) dans notre modèle. FP =670, FN =916, VP =2532, et VN =2213.

	La valeur réelle	Prédicat
VP	1	1
VN	0	0
FP	0	1
FN	1	0

Tableau 14 : Comparaison entre les valeurs réelles et les prédictions pour les différentes classes (VP, VN, FP, FN)

D'après ces résultats, nous avons extrait les tweets et les émojis qui représentent un sentiment positif, négatif. Nous avons également identifié les émojis non connus qui ne sont pas directement liés à un sentiment spécifique. Voici quelques exemples :

- **Exemple pour tweet VN :**

Tweet :

للّه يرحمك بابا مش هيجي زيك ابدا 💔

Justification :

Dans le contexte du cette tweet, le sentiment est négative et l'emoji de cœur brisé indique un sentiment négatif (tristesse et la douleur émotionnelle). Par conséquent, le modèle a classé ce tweet comme un véritable tweet négatif.

- **Exemple pour tweet VP :**

Tweet :

وجزاهم بما صبروا جنة وحريرا لاَ تفقد صبرك 🌸💚 فـ ٱلأشيآء ٱلجميلة تأتي بعد صبر جميل 🌸💚

Justification :

Dans le contexte du cette tweet, le sentiment est Positive et l'emoji de rose et le cœur vert indique un sentiment positif (beauté, de tranquillité et d'espoir). Par conséquent, le modèle a classé ce tweet comme un véritable tweet positif.

- **Exemple pour tweet FN :**

Tweet :

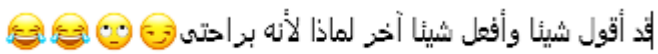
أبوي عارف كيف يروقني أمس زعلانه عليه وجايب لي ورد معه بالليل 😉💞 تاجراسي

Justification :

Dans le contexte de ce tweet, les emojis du sourire narquois et du cœur pulsant expriment une satisfaction et un amour passionné, respectivement. La combinaison de ces emojis crée une ambiance positive. Cependant, malgré cette tonalité positive, le modèle a classé ce tweet comme négatif en raison de la présence du mot "زعلانة" qui exprime un sentiment de tristesse.

- **Exemple pour tweet FP :**

Tweet :



Justification :

Dans ce tweet, on retrouve des emojis tels que le visage avec un sourire narquois, les yeux levés au ciel et le visage qui pleure de rire. Ils expriment une attitude de moquerie et de ridicule. Malgré cette ambiance négative, la présence du visage qui pleure de rire indique une grande hilarité. La répétition de cet emoji renforce l'idée d'un rire incontrôlable. Par conséquent, le modèle a classé ce tweet comme positif, peut-être en interprétant l'accent mis sur le rire comme une réaction positive

- Maintenant, on applique les hyperparamètres performants sur une base de données ne contenant aucun emoji et dans le tableau suivant on comparer les résultats entre l'utilisation d'émojis et l'absence d'émojis.

	Losss train	Loss validation	Accuracy validation	Accuracy train	AUC_ROC
Avec les émojis	0.326801	0.490318	79.20	85.369036	0.87
Sans les émojis	0.321967	0.492049	78.88	85.592385	0.88

Tableau 15 : La comparaison entre les modèles performants (avec et sans émojis)

3.6. La courbe ROC pour le modèle sans emojis

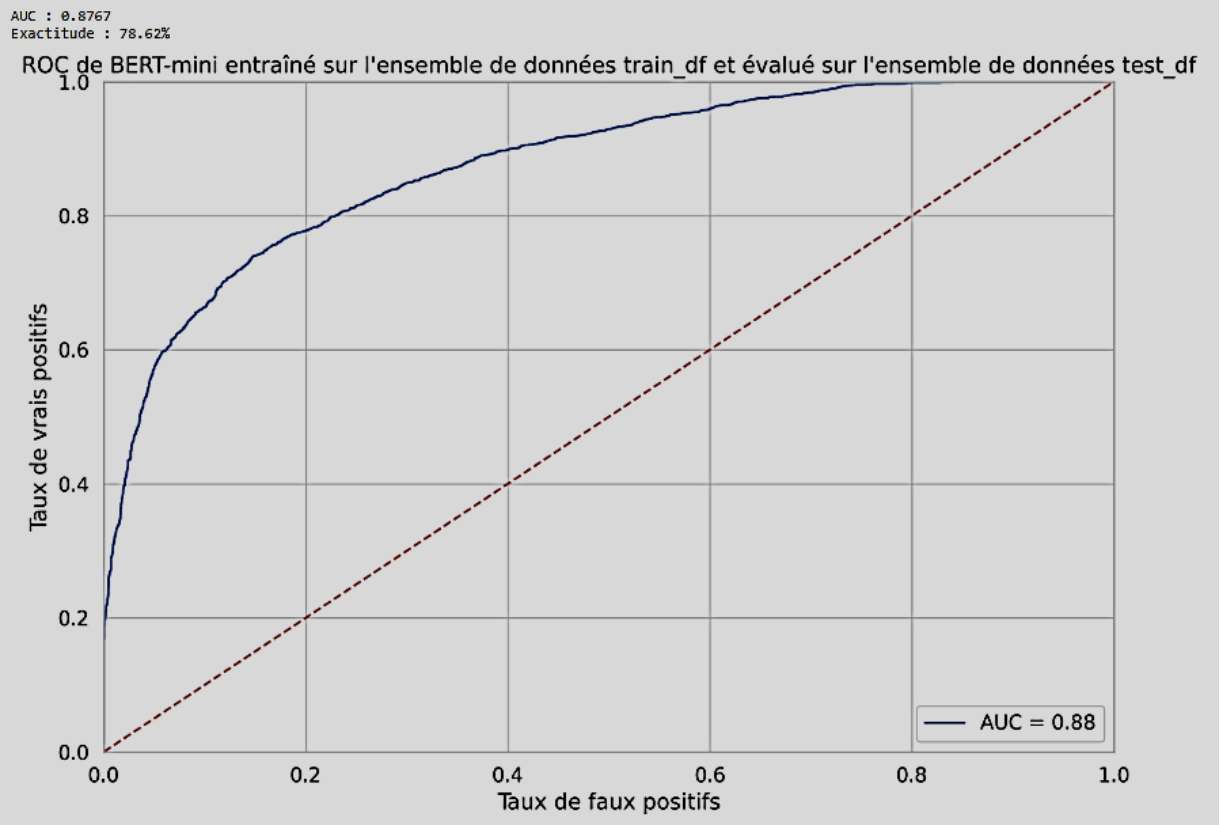


Figure 36 : La courbe ROC pour l'ensemble d'apprentissage et de test sans les emojis

3.7. La matrice de confusion pour le modèle choisi sans émojis

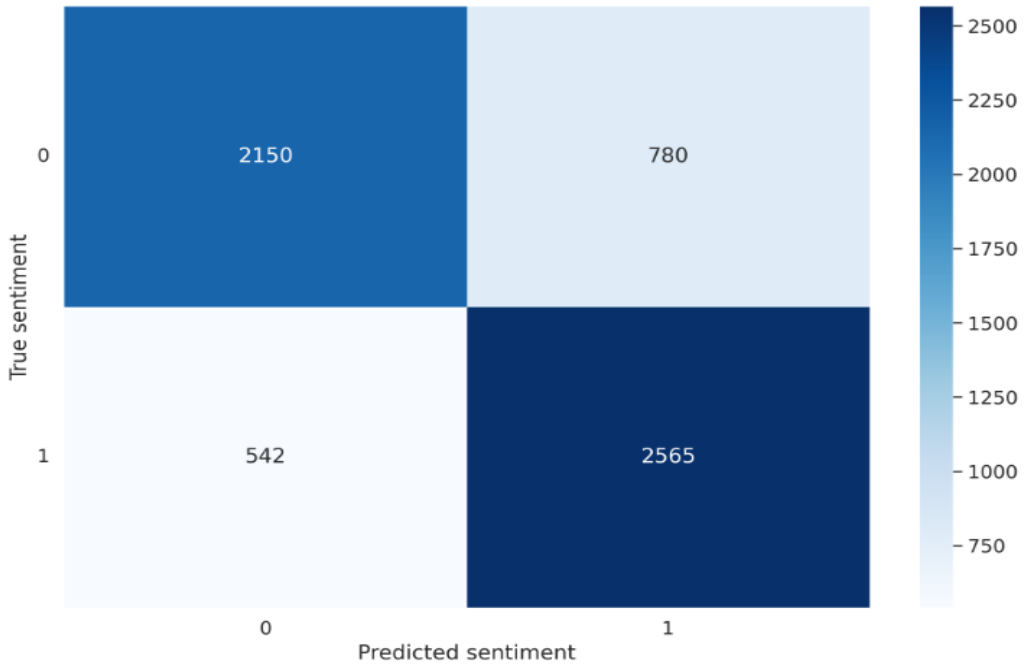


Figure 37 : Matrix de confusion de modèle choisi sans émojis

Donc, d'après les résultats précédents, nous concluons que le modèle obtient de meilleurs résultats lorsqu'il tient compte de la présence des émojis plutôt que de les ignorer.

4. Conclusion

En conclusion, les contributions et expérimentations du modèle pré entraîné BERT ont démontré que prendre en compte la présence des émojis plutôt que les ignorer conduit à de meilleurs résultats. Cela souligne l'importance de considérer ces symboles visuels dans le contexte du langage naturel, car ils apportent des informations précieuses pour la compréhension et l'analyse du texte. Ces résultats ouvrent la voie à l'amélioration des modèles de traitement du langage en intégrant de manière plus efficace les émojis dans les tâches de compréhension et de génération de texte.

Chapitre 04

Implémentation

CHAPITRE 04 : IMPLEMENTATION

1. Introduction

Ce chapitre d'implémentation vise à présenter les aspects clés liés aux ressources matérielles et logicielles, à l'environnement d'exécution, ainsi qu'aux bibliothèques utilisées dans le cadre du développement de notre système. Ce chapitre offre un aperçu détaillé des choix technologiques effectués et met en évidence les décisions prises pour assurer le bon fonctionnement de l'application.

2. Les ressources matérielles et logicielles

2.1. Les ressources matérielles

Les expériences ont été compensées sur une machine virtuelle hébergée dans l'environnement Google Colab, qui présente de bonnes performances. Voici les spécifications de cette machine virtuelle utilisée pour les expériences.

Type	Caractéristiques	Endroit
CPU	Intel(R) Core(TM) CPU 2.60 GHz, RAM 4.00 Go	Local
GPU	GPU T4	A distance

2.2. Les ressources logicielles

2.2.1. Environnement de programmation utilisée

- **Google Colaboratory** :

Colaboratory également connu sous le nom de Colab, est un service basé sur le cloud qui utilise Jupyter Notebooks pour enseigner et mener des recherches sur l'apprentissage automatique. Il offre un environnement prêt à l'emploi pour exécuter des tâches d'apprentissage profond et accorde un accès gratuit à un GPU robuste. Ce document propose un examen détaillé des ressources matérielles fournies par Colaboratory (2316).

Figure 38 : logo de Google colaboratory (Google Collab ou Google Colaboratory : qu'est-ce que c'est)

- **Jupyter**

Les notebooks Jupyter sont la technologie sous-jacente sur laquelle Colab est basé. Jupyter est un outil open source basé sur un navigateur qui intègre des langages interprétés, des bibliothèques et des outils de visualisation. Les notebooks Jupyter peuvent être exécutés localement ou dans le cloud. Chaque bloc-notes se compose de plusieurs cellules, où chaque cellule contient soit un langage de script, soit un code de balisage, et la sortie est intégrée dans le document. Les sorties typiques incluent du texte, des tableaux, des graphiques et des graphiques. L'utilisation de cette technologie facilite le partage et la reproduction des travaux scientifiques, car les expériences et les résultats sont présentés de manière autonome ((Documentation Jupyter Notebooks).

Figure 39 : Logo de Jupyter **(JupyterLab : une interface de bloc-notes de nouvelle génération)**

2.2.2. Les bibliothèques nécessaires

- **Numpay**

NumPy est une bibliothèque Python très populaire qui est principalement utilisée pour effectuer des calculs mathématiques et scientifiques. Elle offre de nombreuses fonctionnalités et outils qui peuvent être utiles pour les projets de Data Science. Se familiariser avec NumPy est une étape indispensable dans un projet de formation en Data Science (NumPy : la bibliothèque Python la plus utilisée en Data Science, 2023).

Figure 40 : Logo de Numpy

- **Pandas**

La bibliothèque logicielle open-source Pandas est spécifiquement conçue pour la manipulation et l'analyse de données en langage Python. Elle est à la fois performante, flexible et simple d'utilisation. Grâce à Pandas, le langage Python permet enfin de charger, d'aligner, de manipuler ou encore de fusionner des données. Les performances sont particulièrement impressionnantes quand le code source back-end est écrit en C ou en Python. Le nom « Pandas » est en fait la contraction du terme « Panel Data » désignant les ensembles de données incluant des observations sur de multiples périodes temporelles. Cette bibliothèque a été créée comme un outil de haut niveau pour l'analyse en Python (Pandas : la bibliothèque Python dédiée à la Data Science).

Figure 41: Logo de Pandas

- **Transformers**

La bibliothèque "Transformers" est une bibliothèque open source largement utilisée dans le domaine du traitement du langage naturel (NLP). Elle prend en charge plusieurs frameworks populaires tels que TensorFlow et PyTorch, permettant aux utilisateurs de choisir le framework qui convient le mieux à leurs besoins. Elle offre également une large gamme de modèles pré-entraînés dans différentes langues (C, 2022).

- **PyTorche**

PyTorch est une bibliothèque d'IA, développée par Meta, écrite en Python pour se lancer dans le deep learning et le développement de réseaux de neurones artificiels. À partir de plusieurs variables, elle peut servir à réaliser des calculs de gradients ou à utiliser des tableaux multidimensionnels obtenus grâce à des tenseurs. PyTorch est livré avec deux primitives. La première, torch.utils.data.DataLoader, est conçue pour créer des mini-lots de données à partir des datasets d'entrainement. La seconde, torch.utils.data.Dataset, permet d'utiliser des sets de données pré chargés (Crochet-Damais A. , 2022).

Figure 42 : Logo de PyTorch **(Pytorch, logo Icon)**

- **Autres bibliothèques utilisées**

Bibliothèques	Description
Watermark	une extension de Jupyter permet d'afficher les informations des versions des package utiliser dans le notebook.
Textwarp	Un module Python intégré qui fournit des fonctions pour envelopper et formater du texte sur plusieurs lignes.
Seaborn	Une bibliothèque pour la visualisation de données statistiques.
Matplotlib	Une bibliothèque pour la création de graphiques et de visualisations.
Sklearn	Une bibliothèque populaire pour l'apprentissage automatique (Machine Learning).
Collections	Un module qui fournit des structures de données supplémentaires par rapport aux types de données intégrés en Python.
Textwarp	Un module pour le formatage et la manipulation de texte.
Langdetect	Une bibliothèque Python qui permet de détecter automatiquement la langue d'un texte donné.
Emoji	Une bibliothèque Emoji permet de manipuler et de travailler avec des Emojis dans du texte.
Wordcloud	Une bibliothèque pour créer des nuages de mots à partir de données textuelles.
re	Un module pour les expressions régulières, utilisé pour le traitement de chaînes de caractères.
os	Un module pour les fonctions liées au système d'exploitation, comme la gestion des fichiers et des répertoires.
csv	Le module Python intégré pour la lecture et l'écriture de fichiers CSV.
unicodedata	Un module Python intégré qui fournit des fonctions pour travailler avec des caractères Unicode.
tqdm	Une bibliothèque pour l'affichage d'une barre de progression lors de l'itération sur les données. Elle permet de visualiser l'avancement des boucles et des opérations longues.
Time	Un module de la bibliothèque standard de Python utilisé pour mesurer le temps d'exécution ou introduire des délais.
Pickel	Le module Python intégré pour la sérialisation d'objets Python en fichiers binaires.

Tableau 16 Tableau de bibliothèques utilisées

3. Extraits du code

Dans cette section on place les extraits de code pertinents pour illustrer des concepts clés de notre étude.

3.1. Partie du code qui réalise le prétraitement

```python
def text_preprocessing(text):

    # Normaliser l'encodage Unicode
    text = unicodedata.normalize('NFC', text)
    # Supprimer les mentions d'entités (noms d'utilisateur)
    text = re.sub(r'@\w+', '', text)
    # Utiliser une expression régulière pour supprimer les lettres répétées
    #text = re.sub(r'(\w)\1+', r'\1', text)
    # Remplacer '&amp;' par '&'
    text = re.sub(r'&amp;', '&', text)
    # Supprimer les URLs
    text = re.sub(r'http\S+|www\S+|https\S+', '', text)
    # Supprimer la ponctuation à l'exception des emojis
    punct = string.punctuation.replace(':', '')
    pattern = r'[{}]'.format(re.escape(punct))
    text = re.sub(pattern, '', text)
    # Supprimer les caractères non alphabétiques sans supprimer les emojis
    emoji_pattern = regex.compile(r'\p{Emoji}', flags=regex.V1)
    emojis = emoji_pattern.findall(text)
    emojis_joined = ' '.join(emojis)
    pattern = r'[^\w\s{}]'.format(re.escape(emojis_joined))
    text = re.sub(pattern, '', text)
```

Figure 43: Fonction de du prétraitement

3.2. Partie du code qui réalise la création des jetons

```python
def preprocessing_for_bert(data, version="mini", text_preprocessing_fn=text_preprocessing):
    # Créer des listes vides pour stocker les résultats
    input_ids = []
    attention_masks = []
    tokenizer = AutoTokenizer.from_pretrained("asafaya/bert-mini-arabic")
    if version == "mini" else AutoTokenizer.from_pretrained("asafaya/bert-base-arabic")

    # Pour chaque phrase...
    for i, sent in enumerate(data):
        encoded_sent = tokenizer.encode_plus(
            text=text_preprocessing_fn(sent),  # Prétraiter la phrase
            add_special_tokens=True,        # Ajouter `[CLS]` et `[SEP]`
            max_length=145,                  # Longueur maximale pour tronquer/remplir
            padding='max_length',        # Remplir la phrase à la longueur maximale
            return_attention_mask=True,     # Retourner le masque d'attention
            truncation=True
```

Figure 44 : Fonction de Toknisation

3.3. Partie du code qui affiche le nuage de tags

```
text_data = ' '.join(train_df['tweet'])
wordcloud = WordCloud(width=800, height=400, background_color='black',
font_path='/content/drive/MyDrive/KFGQPC Uthmanic Script HAFS Regular.ttf').generate(text_data)
plt.figure(figsize=(10, 6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

Figure 45 : Fonction de nuage de tags

3.4. Partie du code qui réalise l'entrainement

```
for epoch_i in range(epochs):
    # Affichage de l'en-tête du tableau des résultats
    print(f"{'Époque':^7} | {'Lot':^7} | {'Perte entraînement':^18} | {'Perte validation':^16}
    | {'Précision validation':^20} | {'Précision entraînement':^22} | {'Écoulé':^9}")
    print("-"*105)
    # Mesure du temps écoulé pour chaque époque
    t0_epoch, t0_batch = time.time(), time.time()
    # Réinitialisation des variables de suivi au début de chaque époque
    total_loss, batch_loss, batch_counts = 0, 0, 0
    train_accuracy = []  # Suivi de la précision d'entraînement
    # Passage en mode entraînement du modèle
    model.train()
```

Figure 46 : Fonction d'entrainement

3.5. Partie du code qui réalise le teste

```
# Run `preprocessing_for_bert` on the test set
print('Tokenizing data...')
test_inputs, test_masks = preprocessing_for_bert(x_test)

# Create the DataLoader for our test set
test_dataset = TensorDataset(test_inputs, test_masks)
test_sampler = SequentialSampler(test_dataset)
test_dataloader = DataLoader(test_dataset, sampler=test_sampler, batch_size=32)
```

Figure 47 : Fonction de test

3.6. Partie du code qui affiche la courbe ROC

```
def evaluate_roc(probs, y_true, model_name, dataset_name, test_dataset_name):
    preds = probs[:, 1]
    fpr, tpr, threshold = roc_curve(y_true, preds)
    roc_auc = auc(fpr, tpr)
    print(f'AUC : {roc_auc:.4f}')

    # Calcul de l'exactitude sur l'ensemble de test
    y_pred = np.where(preds >= 0.5, 1, 0)
    exactitude = accuracy_score(y_true, y_pred)
    print(f'Exactitude : {exactitude*100:.2f}%')

    # Tracer la courbe ROC
    plt.title(f"ROC de {model_name} entraîné sur l'ensemble de données {dataset_name}
    et évalué sur l'ensemble de données {test_dataset_name}")
    plt.plot(fpr, tpr, 'b', label='AUC = %0.2f' % roc_auc)
    plt.legend(loc='lower right')
    plt.plot([0, 1], [0, 1], 'r--')
    plt.xlim([0, 1])
    plt.ylim([0, 1])
    plt.ylabel('Taux de vrais positifs')
    plt.xlabel('Taux de faux positifs')
    plt.show()
```

Figure 48 : La courbe ROC

3.7. Partie du code qui affiche la matrice de confusion

```
def show_confusion_matrix(confusion_matrix):
 hmap = sns.heatmap(confusion_matrix, annot=True, fmt="d", cmap="Blues")
 hmap.yaxis.set_ticklabels(hmap.yaxis.get_ticklabels(), rotation=0, ha='right')
 plt.ylabel('True sentiment')
 plt.xlabel('Predicted sentiment');
```

Figure 49 : Matrice de confusion

3.8. Partie du code qui extrait les émojis des tweets

```
1 emojis_vp = []
2 emojis_vn = []
3 emojis_fp = []
4 emojis_fn = []
5
6 # Fonction pour extraire les emojis d'une chaîne de caractères
7 def extract_emojis(text):
8     emojis = re.findall(r'[^\w\s,]', text)
9     return ''.join(emojis)
10
11 df_vp['emojis'] = df_vp['Tweet'].apply(extract_emojis)
12 df_vn['emojis'] = df_vn['Tweet'].apply(extract_emojis)
13 df_fp['emojis'] = df_fp['Tweet'].apply(extract_emojis)
14 df_fn['emojis'] = df_fn['Tweet'].apply(extract_emojis)
```

Figure 50 : Fonction d'extraction des emojis

4. Conclusion

La mise en œuvre de ce chapitre repose sur une analyse approfondie des ressources matérielles et logicielles, ainsi que sur la configuration de l'environnement d'exécution. Pour garantir le bon fonctionnement du système, plusieurs ressources matérielles ont été utilisées, notamment des serveurs puissants, des dispositifs de stockage à grande capacité et des équipements réseau performants. Ces ressources sont essentielles pour assurer la scalabilité, la disponibilité et les performances nécessaires à l'application.

Conclusion générale

Conclusion générale

Dans le contexte actuel de l'utilisation massive des médias sociaux et des plateformes de communication en ligne, comprendre les sentiments des utilisateurs est devenu essentiel pour divers domaines, tels que le marketing, la veille stratégique et la gestion de la réputation en ligne. Cependant, la détection des sentiments dans la langue arabe se heurte à des difficultés particulières en raison de ses caractéristiques linguistiques spécifiques, notamment les variations dialectales et la diversité des expressions émotionnelles.

Dans ce mémoire, nous avons exploré l'analyse des sentiments dans la langue arabe en utilisant une architecture Transformer et en se basant sur l'utilisation des émojis. La problématique évoquée était la difficulté d'analyser les sentiments dans une langue complexe comme l'arabe, ainsi que la nécessité d'intégrer les émojis dans le processus d'analyse pour une meilleure compréhension des émotions exprimées.

La motivation de cette recherche était donc d'adopter une approche promotrice pour l'analyse des sentiments en arabe en exploitant les capacités des modèles d'architecture Transformer, qui se sont révélées efficaces dans de nombreux domaines de traitement du langage naturel. De plus, en intégrant les émojis, qui sont devenus des symboles universels des émotions dans la communication en ligne, nous avons cherché à améliorer la précision et la pertinence des résultats obtenus.

Pour atteindre cet objectif, différentes techniques ont été mises en œuvre, notamment la préparation d'un corpus de données annotées en sentiments, l'entraînement d'un modèle basé sur une architecture Transformer adapté à la langue arabe, et l'intégration des émojis dans le processus d'analyse. Des expérimentations ont été menées afin d'évaluer les performances du modèle et comparer les résultats avec d'autres travaux existants.

Les résultats obtenus ont démontré que l'approche adoptée a réussi à analyser les sentiments dans la langue arabe de manière précise et fiable. L'utilisation des émojis a permis une meilleure identification des émotions exprimées dans les textes, en offrant des indices supplémentaires pour une compréhension plus approfondie du contexte émotionnel. Cette approche a également montré une certaine adaptabilité aux variations dialectales, ce qui constitue un avantage significatif dans le contexte de la langue arabe.

En somme, ce mémoire a anticipé à l'avancement de l'analyse des sentiments dans la langue arabe en proposant une approche basée sur l'architecture Transformer et l'utilisation des émojis. Les résultats obtenus ouvrent la voie à de nouvelles applications et possibilités dans divers domaines où la compréhension des émotions exprimées dans les textes arabes est cruciale. Cette recherche peut servir de base solide pour des travaux futurs visant à améliorer encore l'analyse des sentiments dans d'autres langues et contextes spécifiques.

Bibliographie

(s.d.). Consulté le 06 16, 2023, sur https://colab.research.google.com/github/jakevdp/PythonDataScienceHandbook/blob/mas

Birjali, M., Kasri , M., & Ben-Hssane, A. (2021). *A comprehensive survey on sentiment analysis: Approaches, challenges and trends* (Vol. 226).

(Documentation Jupyter Notebooks. (s.d.). Consulté le 06 16, 2023, sur https://docs.jupyter.org/

Deep Learning : définition, concept et usages potentiels. (2017, 01 27). Consulté le 05 17, 2023, sur Nuageo: https://nuageo.fr/2017/01/deep-learning-definition-concept-usages/

La Réalité de La Langue Arabe sur Les Réseaux Sociaux et Son Impact Sur La Sécurité Linguistique et L'identité. (2019). Faculté des Lettres et des Langues Université 8 mai 1945 Guelma.

What Is Machine Learning? A Definition. (2022, 03 14). Consulté le 05 10, 2023, sur expert.ai: https://www.expert.ai/blog/machine-learning-definition/

arabe. (2023). Consulté le 05 30, 2023, sur Université de Birmingham : https://www.birmingham.ac.uk/schools/lcahm/departments/languages/sections/lfa/about/arabic.aspx

Les « Transformers » : pourquoi l'Attention est tout ce dont vous avez besoin?. (2023). Consulté le 14 06, 2023, sur akabi: https://akabi.eu/fr/les-transformers-pourquoi-lattention-%F0%9F%A7%A0-est-tout-ce-dont-vous-avez-besoin/

NumPy : la bibliothèque Python la plus utilisée en Data Science. (2023). Consulté le 06 05, 2023, sur datascientest: https://datascientest.com/numpy

Abdaoui, A. (s.d.). Consulté le 05 23, 2023, sur DziriBERT: a Pre-trained Language Model for the Algerian Dialect: https://github.com/alger-ia/dziribert/tree/main/data

Abdelhamid, N. M. (2021). Techniques d'apprentissage automatique pour l'analyse et la fouille des sentiments dans les réseaux sociaux. BISKRA.

Apprentissage par transfert. (s.d.). Consulté le 05 29, 2023, sur LE MAGIT: https://www.lemagit.fr/definition/Apprentissage-par-transfert

Baheti, P. (2021). *The Essential Guide to Neural Network Architectures.*

Barbosa, L. a. (2010). Robust sentiment detection on twitter from biased and noisy data. Dans *Coling 2010: Posters* (pp. 36-44).

Belaidi, N. (2022). *L'apprentissage supervisé : définition et exemples.*

Bouaziz, A. (2013). Catégorisation automatique de news à l'aide de techniques d'apprentissage supervisé. *document PDF , 4*(3).

Bouaziz, D. E., & Khantoul, B. (2020). *Sentiment analysis with deep learning.* (U. O. Bouaghi, Éd.)

Boyd, D. M. (2007). Sites de réseaux sociaux : définition, histoire et érudition . *Journal of computer-mediated Communication*, 210-230.

C, L. (2022). *Bibliothèque HuggingFace - Un aperçu.*

Collobert, R. a. (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. Dans *Proceedings of the 25th International Conference on Machine Learning* (pp. 160–167).

Crochet-Damais, A. (2022). *Perceptron : retour sur l'ancêtre du machine learning.*

Crochet-Damais, A. (2022). *PyTorch : tout savoir sur la bibliothèque de deep learning.*

DataScientest. (s.d.). Consulté le mai 11, 2023, sur https://datascientest-com.translate.goog/introduction-au-nlp-natural-language

DataScientest. (s.d.). Consulté le 05 12, 2023, sur https://datascientest.com/introduction-au-nlp-natural-language-processing

Dechter, R. (1986). LEARNING WHILE SEARCHING IN CONSTRAINT-SATISFACTION-PROBLEMS.

DistilBERT. (s.d.). Consulté le 06 04, 2023, sur Hugging Face: https://huggingface.co/docs/transformers/model_doc/distilbert

Djerrad, M. a. (2021). Analyse des sentiments des tweets liés au Hirak. Bou Arréridj, Université Mohamed el-Bachir el-Ibrahimi.

Duwairi, R. M., Marji, R., & Rushaidat, S. (2014). *Sentiment Analysis in Arabic tweets.*

étienne, C. (2010). Penser la forme des blogs, entre générique et génétique. *Itin éraires. Littérature, textes, cultures*, 23-31.

Farida, M. (2022). *Rôle des émojis dans la communication verbale et non* (Vol. Revue d'anthropologie). Université Ibrahim Chibbot Dely Brahim Alger3.

Farra, N., Abou Assi, R., & M. Hajj, H. (2010). *Sentence-Level and Document-Level Sentiment* (Vol. Proceedings of International Conference on Data Mining Workshops (ICDMW)).

Forsé, M. (2008). Définir et analyser les réseaux sociaux:les enjeux de l'analyse structurale. *Informations sociales*, 10-19.

Forum, i. (Éd.). (2022). *Are emojis and emoticons different?*

Frąckiewicz, M. (2023). *La controverse Chat GPT-4 : Débattre des avantages et des inconvénients de l'IA avancée.*

Géron, A. (2019, 09). Apprentissage automatique pratique avec Scikit-Learn, Keras et TensorFlow. (I. O'Reilly Media, Éd.) Consulté le 05 21, 2023

Goncalves, P. a. (2013). Comparing and combining sentiment analysis methods. Dans *Proceedings of the first ACM conference on Online social networks.*

Google Collab ou Google Colaboratory : qu'est-ce que c'est. (s.d.). Consulté le 06 16, 2023, sur https://www.hwlibre.com/fr/colaboratoire-google/

Habash, N. Y. (2010). *Introduction to Arabic natural language processing. Synthesis Lectures on Human Language Technologies .*

Habash, N., Rambow, O., & Roth, R. (2009). *Mada+ tokan: A toolkit for arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization.* Center for Computational Learning Systems Columbia University.

Hairy, P. (2021). *Les réseaux de neurones récurrents pour les séries temporelles.*

HANY, Y. (2021). *Arabic Sentiment Analysis using Arabic-BERT.*

HANY, Y. (s.d.). *Arabic Sentiment Analysis using Arabic-BERT*. Consulté le 05 15, 2023, sur https://www.kaggle.com/code/yasmeenhany/arabic-sentiment-analysis-using-arabic-bert/input?select=ar_reviews_100k.tsv

Harsha, A. (2023). *The Ultimate Showdown: RNN vs LSTM vs GRU – Which is the Best?*

Hassine, D. A. (s.d.). Les médias sociaux en Tunisie: Typologie et usages.

HEBB, D. O. (1949). *The Organization of Behavior, New York, Wiley & Sons.*

Hochreiter, S., & Schmidhuber, J. (1997). *LSTM CAN SOLVE HARD LONG TIME LAG PROBLEMS.*

Hua, Y., Zhifeng , Z., & Rongpeng , L. (2018). *Deep Learning with Long Short-Term Memory for.*

Ismaili, Z. (2019, 01 28). Apprentissage Supervisé Vs. Non Supervisé. Consulté le 05 21, 2023

Jarry, A. (2017). *4 idées pour aimanter sur les réseaux sociaux.*

Jun Zhao, K. L. (2016). Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. *Computational Linguistics*, 595-598.

JupyterLab : une interface de bloc-notes de nouvelle génération. (s.d.). Consulté le 05 18, 2023, sur Jupyter: https://jupyter.org/

Kaadoud, I. C. (2018). *Reprenons les bases : Neurone artificiel, Neurone biologique.*

Keldenich, T. (2021). *Fonction d'activation, comment ça marche ? – Une explication simple.*

Kharde, V. A., & Sonawane, S. S. (2016). Sentiment Analysis of Twitter Data: A Survey of. *International Journal of Computer Applications*.

Khurana, D. K. (2023, janvier). Traitement automatique du langage naturel : état de l'art, tendances actuelles et défis. *Multimed Tools Appl 82*, 3713–3744.

Kolkur, S., Gayatri, D., & Reena , M. (2015). Study of Different Levels for Sentiment Analysis. *International Journal of Current Engineering and Technology*.

La Toupie. (s.d.). Consulté le 05 05, 2023, sur https://www.toupie.org/Dictionnaire/Opinion.htm

Lendave, V. (2021). *LSTM Vs GRU in Recurrent Neural Network: A Comparative Study.*

López, J. (2023). *Espejel 10 premiers grands modèles de langage qui ont transformé le NLP au cours des 5 dernières années.*

LSTM, Transformers, GPT, BERT : guide des principales techniques en NLP. (s.d.). Consulté le 06 04, 2023, sur https://france.devoteam.com/paroles-dexperts/lstm-transformers-gpt-bert-guide-des-principales-techniques-en-nlp/

Markrting. (s.d.). Consulté le 05 18, 2023, sur https://c-marketing.eu/emoji-le-nouveau-langage-des-emotions/

Mathur, V. (2022). *Réseau neuronal récurrent (RNN) : types et applications.* University of Sunderland.

Maybach, V. (2019). *Le logo de Twitter. L'histoire d'un logo célèbre*.

Merzouk kahina, M. s. (2019). *détection automatiue des sentiments dans les réseaux sociaux.* bouira, bouira, algeria.

Mittal, V. (2017, 10 03). *Top 15 deep learning applications that will rule the world in 2018 and beyond.* Consulté le 05 21, 2023, sur linkedin: https://www.linkedin.com/pulse/top-15-deep-learning-applications-rule-world-2018-beyond-mittal

Mohammed, M. A., Karrar , H. A., Begonya , G.-Z., Salama, A. M., & Mashael , S. M. (2020). *Un Compréhensife Enquête sur Machine LearFonctionnalitéeMéthodes d'extraction et de classification pour laDiagnostic de COVID-19 basé sur des images radiographiques.*

Moore, T. (2021). *Intelligence artificielle vs Machine Learning vs Deep Learning (IA vs ML vs DL).*

MORERE, Y. (2021). *Les Réseaux de Neurones Récurrents.*

Mouhoubi azzedine mounir, g. m. (2020, nouvembre 11). Analyse de sentiments dans la langue arabe en utilisant différentes d'approches. blida, algeria.

Noam , S., Parmar, N., & Vaswani, A. (2017). *Attention Is All You Need.*

Pak, A. (2012). *Automatic, adaptive, and applicative sentiment analysis.* Université , Paris.

Pandas : la bibliothèque Python dédiée à la Data Science. (s.d.). Consulté le 06 07, 2023, sur datascientest: https://datascientest.com/pandas-python-data-science

Patterson, J. a. (2017). *Deep learning: A practitioner's approach.* O'Reilly Media, Inc.

Poirier, D., Fessant, F., Bothorel, C., Émilie , G. N., & Boullé, M. (2009). *Approches Statistique et Linguistique Pour la Classification* (Vol. Revue des Nouvelles Technologies de l'Information).

Pytorch, logo Icon. (s.d.). Consulté le 06 10, 2023, sur Icon: https://icon-icons.com/icon/pytorch-logo/169823

R, M., M, S., & Z, S. (2021). *Une Nouvelle approche basée deep.*

Rawlings, A. (2018). *Why emoji mean different things in different cultures.*

Réseaux neuronaux convolutifs. (s.d.). Consulté le 06 12, 2023, sur IBM: https://www.ibm.com/fr-fr/topics/convolutional-neural-networks

Risser-Maroix, O., & Kerboua-Benlarbi, S. (2018). *Agents conversationnels pour la recherche d'information. (Traduction de requêtes en langage naturel vers mots clefs).*

Rj, S. (2020). *Artificial intelligence: In modern dentistry* (Vol. De Boeck Supérieur).

Ropars, F. (2018, 01 09). Emoji : définition, histoire et usages. Consulté le 05 09, 2023, sur https://www.blogdumoderateur.com/emoji-definition-histoire-usages/

Rosen, A. (2017, nouvember 07). Tweeting Made Easier.

Rosenblatt, F. (1957). *Cornell Aeronautical Laboratory .*

Saint-Cirgue, G. (2019, 07 02). Apprentissage supervisé : Introduction. Récupéré sur https://machinelearnia.com/apprentissage-supervise-4-etapes/

Sharma, S., Sharma , S., & Anidhya , A. (2020). *FONCTIONS D'ACTIVATION DANS LES RÉSEAUX DE NEURONES.* Revue internationale des sciences appliquées et de la technologie de l'ingénierie.

Shrivastav, A. (2023). *Unité linéaire d'erreur gaussienne (GELU).*

Swarte, M. S. (2023). *Analyse de Sentiment : guide des bonnes pratiques.*

Team, D. S. (2020). *Introduction à l'apprentissage automatique pour les débutants.*

Turpin, A. (s.d.). *L'algorithme Fully connected neural network.*

VERSTEEGH, K. (1997). *The Arabic Language.* Columbia University Press.

Viennet, E. (2008). *Recherche de communautés dans les grands réseaux sociaux.*

Welter, H. (2023). *Deep learning : définition, fonctionnement et applications.*