République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad DAHLAB – Blida1

Faculté des science Département d'Informatique

Mémoire présenté par :

Mlle. SERDOUK Lilia et M. BENAMMOUR Abdelhadi

Pour l'obtention du diplôme de Master

Domaine : Mathématique et Informatique

Filière : Informatique

Spécialité : Traitement Automatique de la Langue

Sujet :

Proposition d'une approche Deep Learning pour l'Analyse des sentiments des Feedbacks des Clients Djezzy exprimés en dialecte Algérien

TEBBI Hanane	Université de Blida 1	Président
KAMECHE Abdellah	Université de Blida 1	Examinateur
BENATMANE Mohamed	Djezzy, OTA Algérie	Encadreur
MEZZI Melyara	Université de Blida 1	Promotrice

Résumé

Le but de cette étude était de proposer une approche Deep Learning pour l'Analyse des sentiments des Feedbacks des Clients Djezzy exprimés en dialecte Algérien. Dans ce travail, nous avons essayé de classifier les sentiments en trois classes (Positif/ Négatif/ Neutre) à l'aide d'un Dataset collecté automatiquement à partir de YouTube et manuellement à partir de Facebook. Pour atteindre nos objectifs, nous avons utilisé des modèles d'apprentissage profond de deux types : ceux basés sur l'apprentissage à partir de zéro qui sont : CNN_Lstm, Bi-Lstm, CNN et ceux basés sur l'apprentissage par transfert qui sont : DZIRI-Bert et Ara-Bert.

Après les tests, les modèles se basant sur l'apprentissage par transfert ont donné les meilleurs résultats. Tel que le meilleur modèle est celui de DziriBert avec un taux d'Accuracy égale à 83% et qui est très encourageant pour notre cas d'étude. Ce résultat peut toutefois être amélioré en augmentant la taille des données.

Mots clés : Analyse des Réseaux Sociaux, Analyse des Sentiments, Apprentissage en Profondeur, Dialecte Algérien.

Abstract

The aim of this study was to propose a Deep Learning approach for the Sentiment Analysis of Feedback from Djezzy Customers expressed in the Algerian dialect. In this work, we tried to classify feelings into three classes (Positive / Negative / Neutral) using a Dataset collected automatically from YouTube and manually from Facebook. To achieve our objectives, we used deep learning models of two types: those based on learning from scratch which are: CNN_Lstm, Bi-Lstm, CNN and those based on transfer learning which are: DZIRI-Bert and Ara-Bert.

After the tests, the models based on transfer learning gave the best results. As the best model is that of DZIRI-Bert with an Accuracy rate equal to 83% and which is very encouraging for our case study. This result can however be improved by increasing the size of the data.

Keywords: Social Network Analysis, Sentiment Analysis, Deep Learning, Algerian Dialect.

ملخص

كان الهدف من هذه الدراسة هو اقتراح نهج التعلم العميق لتحليل المشاعر لردود الفعل من عملاء جيزي المعبر عنها باللهجة الجزائرية. في هذا العمل، حاولنا تصنيف المشاعر إلى ثلاث فئات (إيجابي / سلبي / محايد) باستخدام مجموعة بيانات تم جمعها أوتوماتيكيا من YouTube ويدويًا من Facebook.

لتحقيق أهدافنا، استخدمنا نماذج التعلم العميق من نوعين: تلك القائمة على التعلم من الصفر وهي: CNN_Lstm وBi-Lstm وCNN وتلك القائمة على التعلم الانتقالي وهي: DZIRI-Bert وAra-Bert.

بعد الاختبارات، أعطت النماذج المعتمدة على التعلم بالنقل أفضل النتائج. بحيث أن أفضل نموذج هو نموذج DZIRI-Bert بمعدل دقة يساوي 83٪ وهو أمر مشجع للغاية بالنسبة لدراسة الخاصة بنا. ومع ذلك، يمكن تحسين هذه النتيجة عن طريق زيادة حجم البيانات.

الكلمات المفتاحية: تحليل الشبكة الاجتماعية، تحليل المشاعر، التعلم العميق، اللهجة الجزائرية.

Dédicaces

Avec l'expression de ma reconnaissance, je dédie ce modeste travail a ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leurs exprimer mon sincère gratitude.

A mon cher père, mon précieux offre du dieu.

A mon adorable mère qui a souffert Sans me laisser souffrir.

A mes chers frères et sœurs : Samir, khaoula, Wassim et ma jumelle Rania qui n'ont cessé de me conseiller, encourager et soutenir tout au long de mes études.

A mes belles sœurs mais plutôt mes grandes sœurs Amina et Sarah pour leurs soutiens et pour les bons moments passés ensemble.

Une dédicace toute particulière à mes petits anges rida et Yara et ma nièce Méline.

Sans oublier mon binôme Abdelhadi pour son soutien moral, sa patience et sa compréhension tout au long du projet.

Que ce travail soit l'accomplissement de vos vœux tant allégués, et le fuit de votre soutien infaillible, Merci d'être toujours là pour moi.

Serdouk Lilia

Dédicaces

Tout d'abord je remercie Dieu de m'avoir donné la force et la volonté de mener à bien ce travail.

Je dédie ce travail A mes chers parents, pour tous leurs sacrifices, leurs encouragements et leurs prières tout au long de mes études.

A mon frère Salahedinne et ma sœur Meriem, à mes grands-parents, ainsi qu'à toute ma famille Merci d'être toujours là pour moi.

Benammour Abdelhadi

Remerciements

Nous tenons à remercier toutes les personnes qui ont contribué au succès de notre stage et qui nous ont aidés. Lors de la rédaction de ce mémoire.

Nous voudrons tout d’abord adresser toute notre reconnaissance à notre promotrice et enseignante Madame Mezzi, pour sa patience, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter notre réflexion ainsi pour la qualité de son enseignement. Qu’elle trouve ici l'expression de notre sincère gratitude.

Un grand merci également à Mr Amine Benathmane pour son encadrement durant notre travail et de nous faire découvrir le milieu professionnel.

Nous remercions également toute l’équipe pédagogique de l’université de Saad Dahleb spécialement le département de l’informatique et les intervenants professionnels responsables de notre formation, pour avoir assuré la partie théorique de celle-ci.

Table des matières

La langue arabe et le dialecte algérien et les notions fondamentales

Travaux connexes

Conception et modélisation de la solution

implémentation de la solution

Liste des figures

Liste des tableaux

Liste d’acronymes

TAL : Traitement Automatique du Langage

DL: Deep Learning

CNN: convolutional neural network

Lstm: Long short-term memory

Bi-Lstm: Bidirectional long-short term memory

RNN : recurrent neural network

IA : artificiel intelligence

DALG : algerian dialecte

MSA : Modern Standard Arabic

Bert : Bidirectional Encoder Representations From Transformers

Introduction générale

1. Contexte général

Depuis l'apparition des réseaux sociaux, des millions de personnes partagent leur impression de leurs quotidiens avec leur communauté d'amis et réseau de connaissances. Ces plateformes de communication sont également utilisées par des entreprises, des chercheurs et des universitaires qui s'intéressent à l'opinion des personnes. Cependant, la croissance active des opinions sur les réseaux sociaux a conduit aux intéressés du domaine vers l'impossibilité au traitement manuel des grandes quantités de données et donc les as poussées à l'utilisation de méthodes modernes avec l'apparition de l'intelligence artificiel qui a permis aux entreprises et chercheurs a automatisé leurs processus d'analyse de sentiment en temps réel.

L'analyse de sentiment est un domaine qui est toujours en évolution mais confronté à plusieurs défis, ces défis sont parfois des obstacles pour analyser la polarité exacte du sentiment puisque la machine doit être formée afin de comprendre des particularités telles que les émotions, le sarcasme, les préjugés, la compréhension des nuances …comme le fait le cerveau humain. Mais bien qu'avec les bonnes techniques de traitement du langage naturel ces exceptions ont été surmontées avec des bonnes solutions et qui se sont répondu dans de nombreuses langues et leurs dialectes.

En ce qui concerne le dialecte algérien, il avait reçu peu d'attention et encore moins en écriture latine (Arabizi), mais actuellement on assiste à un intérêt croissant auprès des entreprises et chercheurs vu l'augmentation du volume des textes arabes dans les médias sociaux. Et c'est l'un des dialectes les plus recensé vu l'absence des ressources et du traitement standard comparé aux autres langues et dialectes tel que la représentation linguistique, lexique et syntaxiques comparé à la langue arabe classique.

Tous ces aspects rendent les solutions du traitement du langage naturel qui ont été développé pour le traitement des langues arabes insuffisantes devant un tel dialecte. Cependant beaucoup de recherches et travaux ont été développé avec efficacité pour répondre aux problématiques de ce dernier.

Afin d'effectuer ces analyses et travaux, les entreprises reçoivent chaque jour de gros volumes de commentaires électroniques de la part de leurs clients qui sont sous forme de suggestions, de critiques et aussi recommandation. Ces commentaires sont appelés

« Feedback clients », ils consistent un avantage aux entreprises en identifiant les besoins de leurs clients pour qu'ils puissent améliorer la satisfaction et réduire le taux de désabonnement.

2. Problématique

L'analyse des sentiments est un domaine en plein développement en raison de ses nombreuses applications et a pour but d'analyser les avis des utilisateurs afin d'avoir leur opinion. Mais son application sur les commentaires des clients de l'entreprise Djezzy en dialecte algérien ainsi que le grand volume de données obtenues chaque mois nécessite un bon outil et technique qui permet d'analyser efficacement les opinions de leurs clients.

3. Objectifs

Pour résoudre ce problème nous allons élaborer une stratégie pour l'analyse des sentiments exprimés en dialecte algérien.

Afin d'atteindre cet objectif nous allons effectuer un prétraitement convenable au dialecte algérien ainsi que pour le contexte du domaine de télécommunication.

Nous allons aussi réaliser des modèles capables d'analyser et extraire efficacement la polarité d'un commentaire.

Réaliser ces solutions sous forme d'un outil d'analyse simple en termes d'utilisation afin que les employés du service Customer care puissent bénéficier aisément.

4. Organisation du mémoire

Afin de répondre aux objectifs susmentionnés, nous avons organisé notre mémoire en 4 chapitres :

- Le premier chapitre : Analyse des sentiments : Nous présenterons l'entreprise Djezzy, les notions générales de l'analyse des sentiments.
- Le deuxième chapitre : La langue Arabe et le dialecte algérien et les notions fondamentales : Nous présenterons la langue arabe et le dialecte algérien ainsi que leurs défis et particularités et les notions fondamentales.
- Le troisième chapitre : Travaux connexes : Nous présenterons les travaux qui ont déjà été fait dans le domaine.

- Le quatrième chapitre : Conception et modélisation de la solution : Nous allons expliquer notre architecture.
- Le cinquième chapitre : Implémentation de la solution : Nous présentons l'implémentation des différentes tâches de notre système ainsi que les résultats des modèles formés pour le dialecte algérien dans le contexte du domaine de télécommunication.

Chapitre I : ANALYSE DES SENTIMENTS

1. Introduction

L'analyse de sentiments vise à déterminer l'attitude d'un écrivain envers certains sujets ou la polarité globale des sentiments d'un texte, comme positif ou négatif, ou neutre [1], mais elle peut aussi déterminer la polarité pour détecter des sentiments et des émotions spécifiques comme la colère, la joie, la tristesse, etc. Ou encore l'urgence (urgent, pas urgent), les intentions (intéressé v. pas intéressé), et parfois même des caractéristiques psychologiques des utilisateurs d'une certaine plateforme (exemple : profile criminel, comportement raciste…etc.).

Dans ce chapitre, nous allons faire le tour sur la notion de l'Analyse des Sentiments en nous intéressons à ses origines, ses applications, ses types, et ses niveaux entre autres.

2. Présentation de l'entreprise Djezzy

Djezzy est la marque commerciale retenue pour représenter le réseau GSM, 3G et 4G d'Optimum Télécom Algérie spa, ce terme vient de la contraction de deux mots : El Djazaa (le cadeau) et El Djazair (l'Algérie). Grâce à des contrats Roaming avec 458 opérateurs dans 158 pays, l'entreprise compte près de 16 millions d'abonnés joignable partout dans le monde et qui ont fait d'elle le numéro un en Algérie.

OTA fait preuve de proximité avec ses clients en mettant à leur disposition 136 centres de services, et un centre d'appels joignable 24H/24, 7J/7, sur tout le territoire national. L'entreprise dispose de plusieurs sièges à Alger et de deux centres régionaux (Est et ouest).

Outre la qualité de ses services, OTA dispose d'une richesse humaine de plus de 2800 employés, qui représentent le vrai secret de sa réussite.

2.1.Historique

OTA, opérateur de télécommunications algérien a été créé en juillet 2001. Leader dans le domaine de la téléphonie mobile et des technologies de communications numériques, l'entreprise fournit une vaste gamme de services tels que le prépayés, le post-payé, le Data ainsi que les services à valeur ajoutée VAS et le SUT.

En janvier 2015, le Fonds National d'Investissement (FNI) prend le contrôle de 51% du capital de la société alors que le partenaire étranger, le Groupe VEON garde la responsabilité du management de l'entreprise.

OTA couvre 95 % de la population à travers le territoire nationale et ses services 3G sont déployés dans les 48 wilayas depuis fin 2016. OTA a lancé ses services 4G le 1er octobre 2016 et couvre 28 wilayas au 31 décembre 2018 avec l'engagement de couvrir plus de 50% de la population à l'horizon 2021.

OTA fait partie du groupe VEON (coté à Nasdaq et Euronext), une entreprise de communication et de technologie internationales guidée par une vision construite sur des racines entrepreneuriales et dont les valeurs sont basées sur la satisfaction du client, l'innovation, le partenariat et la transparence.

Avec plus de 2.5 milliards de dollars d'investissement depuis 2001 à ce jour, fort d'un capital humain de 2800 employés et plus de 16 millions d'abonnés, OTA demeure, en tous points de vue, l'opérateur préféré des Algériens

2.2. Missions

Afin de réaliser ses objectifs, OTA a pour mission de :

- Offrir les meilleurs produits, de qualité, à des prix compétitifs.
- Déployer des infrastructures à la pointe de la technologie.
- Créer pour ses employés le meilleur environnement de travail et d'épanouissement.
- Contribuer activement au bien-être des Algériens.
- Optimiser la création de valeur pour ses actionnaires, à travers un contrôle strict des coûts.
- Appliquer rigoureusement sa politique environnementale.
- Améliorer sans cesse ses processus internes dans le respect de sa politique qualité.

2.3. Valeurs

Les valeurs de Djezzy se résument en :

- **Satisfaction client :** Ecouter, apprendre et réussir.
- **Innovation** : Constamment progresser vers une expérience client exceptionnelle.
- **Esprit d'entrepreneur :** Être agile pour saisir les opportunités et donner vie à toutes les perspectives.
- **Collaboration :** Travailler en équipe, apprendre de ses erreurs pour générer valeur et succès
- **Intégrité :** Défendre avec fermeté les normes d'éthique et d'intégrité les plus strictes.

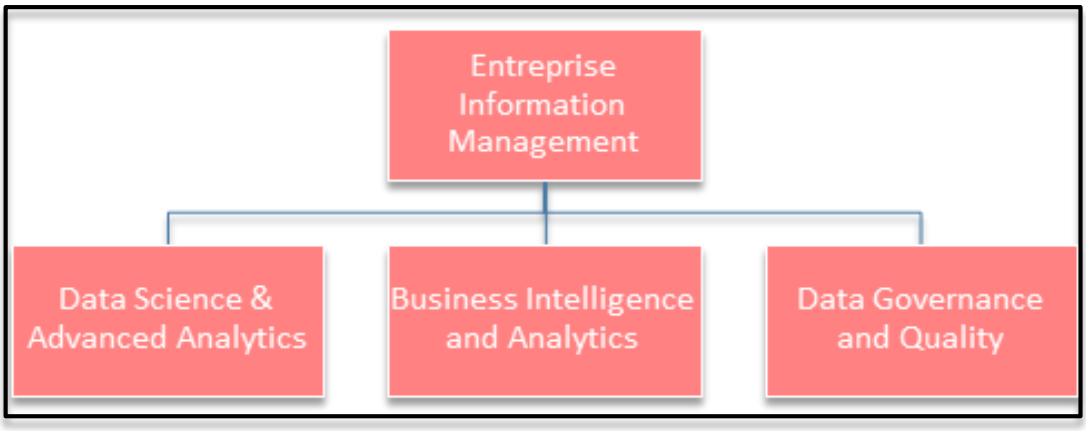


Figure 1:Structure d'accueil de l'entreprise Djezzy

3. Contexte global

L'étude de l'opinion publique peut nous fournir des informations précieuses. L'analyse de sentiments sur les réseaux sociaux, tels que Twitter ou Facebook, est devenue un moyen puissant pour connaître les opinions des utilisateurs et dispose d'un large éventail d'applications telles que [2] :

- **La Surveillance des médias sociaux :**

  Les publications sur les réseaux sociaux contiennent souvent des opinions honnêtes sur les produits, services et entreprises, car elles ne sont pas sollicitées par exemple : « J'adore l'interface utilisateur. L'installation a pris cinq minutes et nous étions prêts à commencer »

- **Service client :**

  La gestion du support client présente de nombreux défis en raison du grand nombre de demandes, des sujets variés et des diverses branches au sein d'une entreprise - sans parler de l'urgence d'une demande donnée. L'analyse des sentiments avec compréhension du langage naturel (NLU pour Natural Language Understanding en Anglais) lit le langage humain et son comportement tel que le sens, l'émotion, le ton, etc., afin de comprendre les demandes des clients, comme le ferait une personne.

- **Veille de la marque et gestion de la réputation :**

  La surveillance de la marque est l'une des applications les plus populaires de l'Analyse de Sentiments dans les entreprises. Les mauvaises critiques peuvent faire boule de neige en ligne, et plus vous les laissez longtemps, plus la situation sera grave. Grâce à l'Analyse de sentiment, les clients seront immédiatement informés des mentions négatives de la marque. De plus, ils pourront suivre l'image et la réputation de leur marque au fil du temps ou à tout moment, afin de suivre leurs progrès.

- **Recherche de marché et de concurrents :**

  L'utilisation de l'Analyse des Sentiments peut servir pour les études de marché et des concurrents. Elle permet de découvrir qui reçoit des mentions positives parmi les concurrents et de comparer les efforts de marketing.

  Par ailleurs, elle permet d'analyser le langage positif que les concurrents utilisent pour parler à leurs clients Et intègre une partie de ce langage dans son propre message de marque afin de bénéficier de la préférence du consommateur.

4. Origine de l'Analyse de Sentiments

L'origine de l'Analyse des Sentiments remonte dans les études sur l'Analyse de l'opinion publique au début du XXe siècle aux alentours de 1950, lorsque l'Analyse des sentiments était principalement utilisée sur des documents papier écrits et dans l'analyse de la subjectivité du texte effectuée par la communauté de la linguistique computationnelle dans les années 1990 [3]. Cependant, on considère que l'Analyse de

Sentiment a réellement commencé au début des années 2000 avec les articles [1]publiés [2]par Bo Pang [3]et Liliane Lee [4]où l'éclosion de l'Analyse des Sentiments par ordinateur ne s'est produite qu'avec la disponibilité de textes subjectifs sur le Web 2.0 d'où l'intérêt va croissant pour connaître les opinions des internautes qui s'y expriment spontanément et en temps réel [4].

5. Les applications d'Analyse des Sentiments

De nos jours, l'Analyse des Sentiments ne se limite pas à une seule application, Elle est devenue un outil indispensable dans plusieurs domaines afin d'aider à la prise de décision [5].

La figure ci-dessous représente un Tweet qui indique l'appréciation d'un client sur le service de la société d'Amazon.

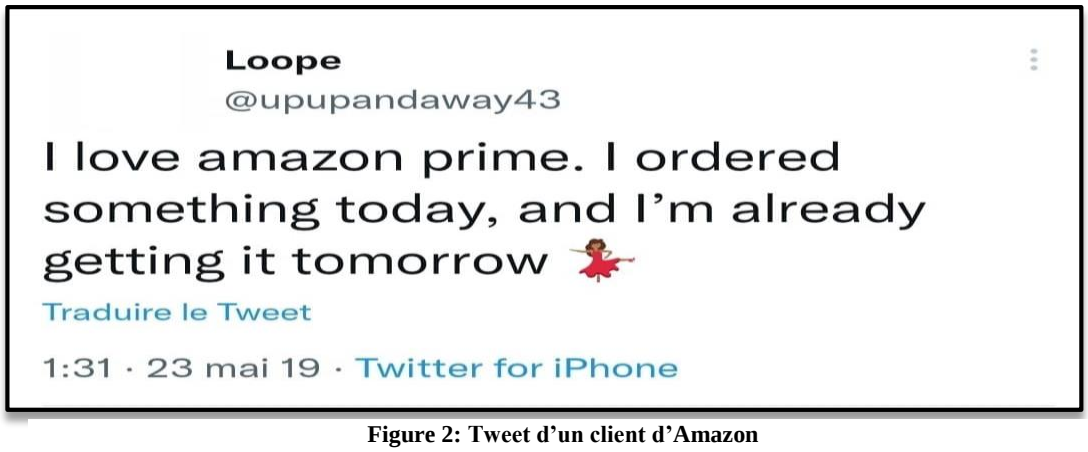


Figure 2: Tweet d'un client d'Amazon

Pour les entreprises, les commentaires des clients jouent un rôle extrêmement important, où ils peuvent aider à prendre les mesures appropriées pour améliorer leurs produits ou services et leur stratégie commerciale. Elle pourrait même sensibiliser à la sécurité des données et au danger de failles de sécurité.

Dans le domaine de la Politique, L'Analyse des Sentiments peut être utilisée pour prédire les élections, où elle montre que les données analysées sous formes de sondage à partir du réseau social Twitter sont plus fiables que celles de la plate-forme

[1] https://arxiv.org/abs/cs/0409058
[2] https://pastel.archives-ouvertes.fr/tel-00408754/document
[3] **Bo Pang :** Chercheur scientifique chez Google
[4] **Liliane Lee :** Professeur d'informatique à l'Université Cornell

des élections. A tel point qu'elle ait acquis le potentiel de devenir une plate-forme capable de rivaliser avec la télévision et la radio.

Par ailleurs, L'Analyse de Sentiments ne se limite pas juste à ces domaines là, mais bien plus, tel que le domaine sportif, éducatif entre plein d'autres. Telle qu'elle peut de nos jours construire ou défaire des réputations ainsi que former et déformer des courants publics.

6. Les niveaux de l'analyse des sentiments

L'analyse des sentiments peut se faire sur trois niveaux. Le schéma ci-après résume ses différents niveaux d'Analyse [6]

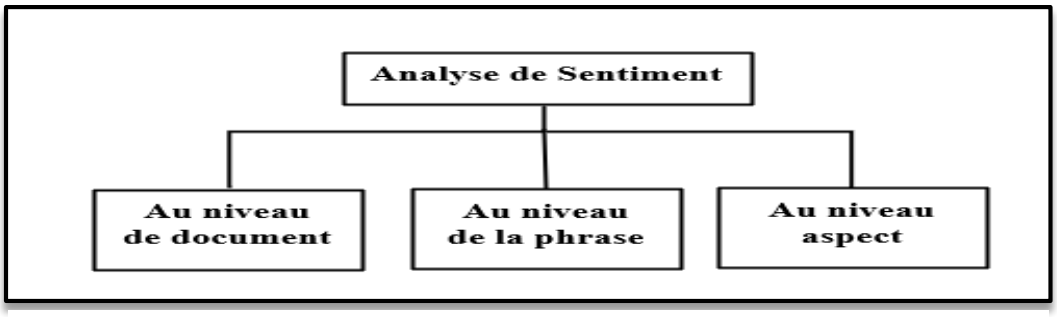


Figure 3: les niveaux d'Analyse de Sentiment [7]

6.1. Au niveau du document

La tâche à ce niveau, consiste à classer le sentiment pour le document. Le document est considéré sur un seul sujet. Ainsi, les textes qui comprennent un apprentissage comparatif ne peuvent pas être considérés sous ce niveau.

Texte	Sentiment
Cette offre est excellente	Positif

Tableau 1: Exemple de sentiment au niveau du document

6.2. Au niveau de la phrase

La tâche à ce niveau s'intéresse aux phrases. Elle consiste à déterminer si chaque phrase exprime une opinion positive, négative ou neutre. Si une phrase n'a aucune opinion cela signifie qu'elle est neutre. Ce niveau d'analyse est lié à la classification de la subjectivité.

Texte	Sentiment
Cette offre est excellente, le téléphone capte bien le réseau	Positif

Tableau 2: Exemple de sentiment au niveau de la phrase

6.3.Au niveau aspect

Le niveau d'aspect donne une analyse détaillée, La tâche principale du niveau entité est d'identifier l'aspect du texte. Aussi les comparatifs font également partie de l'analyse des sentiments à ce niveau.

Texte	Aspect	Sentiment
Le téléphone est excellent, mais il faut encore travailler sur la durée de vie de la batterie et les problèmes de sécurité	Le téléphone	Positif
	La durée de la vie de la batterie	Négatif
	La sécurité	Négatif

Tableau 3.Exemple de sentiment au niveau de l'aspect

7. Les défis de l'analyse des sentiments

Si l'analyse de sentiment est un domaine en constante évolution, ceci est dû aux nombreux défis qu'elle représente. En effet, le langage humain est souvent vague ou très contextuel, ce qui rend la compréhension automatique très difficile sans aide humaine. L'aide humaine est essentielle lors de la formation en apprentissage automatique d'une solution d'analyse de sentiment. Dans cette section, nous allons présenter quelques-uns des défis liés à l'Analyse de sentiments [8].

7.1.Extraction d'entités nommées

Les entités nommées sont des expressions nominales définies qui font référence à des types spécifiques d'individus, tels que des organisations, des personnes, des dates, etc. Le but de l'extraction d'entités nommées est d'identifier toutes les mentions textuelles des entités nommées dans un morceau de texte.

7.2.Extraction d'informations

L'information se présente sous de nombreuses formes et tailles. La complexité du langage naturel peut rendre très difficile l'accès aux informations contenues dans le texte d'opinion.

7.3.Détermination des sentiments

La détermination du sentiment est une tâche qui attribue une polarité de sentiment à un mot, une phrase ou un document. Une manière traditionnelle d'attribuer la polarité des sentiments consiste à utiliser le lexique des sentiments. Les adjectifs d'une phrase sont importants dans l'exploration d'opinion car ils ont plus de probabilité de transporter des informations lorsque le problème d'analyse des sentiments est pris en compte.

7.4.Résolution de coréférence

La résolution des coréférences doit être effectuée au niveau de l'aspect et au niveau de l'entité. Dans le cas du texte opiniâtre, on peut voir des textes comparatifs. Ces textes comparatifs peuvent contenir des coréférences. Ces références doivent être efficacement résolues pour produire des résultats corrects.

7.5.Extraction de relations

L'extraction de relations consiste à trouver la relation syntaxique entre les mots d'une phrase. La sémantique d'une phrase peut être trouvée en extrayant les relations entre les mots et cela peut être fait en connaissant les dépendances des mots.

7.6.Dépendance de domaine

Un classificateur de sentiments formé pour classer les polarités d'opinion dans un domaine peut produire des résultats médiocres lorsque le même classificateur est utilisé dans un autre domaine. Le sentiment s'exprime différemment selon les domaines. Par exemple, considérons deux domaines, l'appareil photo numérique et la voiture. La façon dont les clients expriment leurs pensées, leurs points de vue sur les appareils photo numériques seront différents de celle des voitures. Mais certaines similitudes peuvent également être présentes. L'analyse des sentiments est donc un problème qui dépend fortement du domaine. Par conséquent, l'analyse des sentiments inter-domaines est un problème difficile qui doit être résolu.

8. La classification de la Subjectivité et de l'objectivité

Cette tâche est généralement définie comme la classification d'un texte donné (généralement une phrase) dans l'une des deux classes suivantes [9] : **objective** qui Présente des informations factuelles sur le monde comme : "iPhone est un produit Apple" ou **subjective** qui exprime des sentiments personnels, des opinions ou des croyances comme : "J'aime l'iPhone".

Son objectif principal est de diviser l'ensemble des documents ou des phrases en l'une des deux classes : **objective** ou **subjective**. En utilisant l'analyse des sentiments, nous pouvons extraire des phrases subjectives. Les informations factuelles générées par des phrases objectives doivent être supprimées. Lorsqu'elles contiennent des émotions ou des sentiments, les phrases subjectives sont importantes pour la procédure de l'analyse des sentiments. Les phrases subjectives sont composées des vues suivantes : des utilisateurs, des perspectives, des pensées, des commentaires et des opinions sur le niveau de la phrase [10].

La figure 4 récapitule la classification dont nous venons de parler.

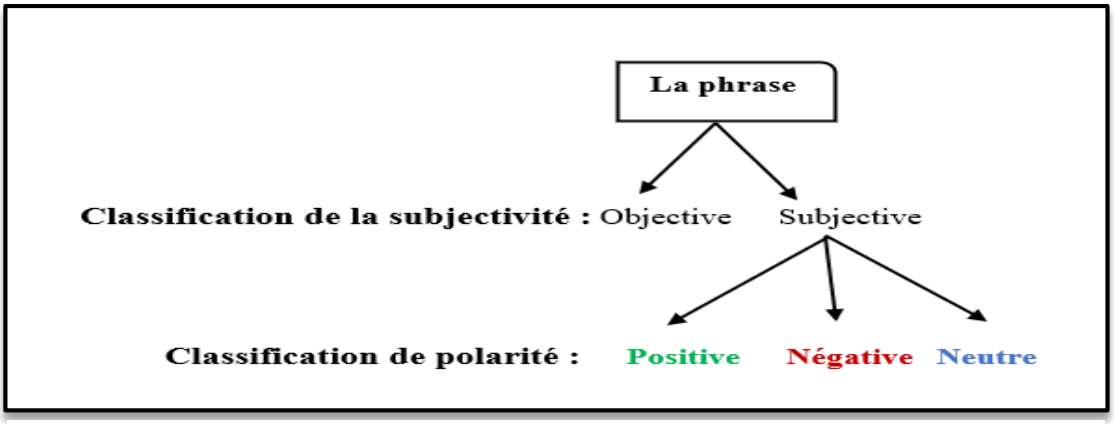


Figure 4: Une classification hiérarchique générale des phrases du point de vue des sentiments [11]

9. L'opinion

Dans cette section, nous allons parler d'une notion incontournable à la compréhension de ce chapitre qui est la notion d'**Opinion** qui englobe l'Analyse des Sentiments dans une dimension plus large.

9.1.Définition de l'opinion

Les opinions peuvent représenter des avis, sentiments, évaluations, attitudes, émotions des personnes envers des entités, individus, problèmes, produits, évènements, sujets, … et leurs attributs. Cette tâche est devenue très utile. Par exemple, les entreprises cherchent toujours à trouver les opinions du public ou de leurs consommateurs au sujet de leurs produits et services. Les clients potentiels veulent aussi connaître les avis des utilisateurs existants avant d'utiliser un service ou d'acheter un produit. Les opinions peuvent être exprimées sur diverses choses. Par exemple : un produit, un service, une personne, une organisation, un événement ou un sujet et ce, par n'importe qui (personne ou organisation, voire même état). On utilise le terme **Entité** pour désigner l'objet cible qui a été évaluée. Une entité et est un produit, service, personne, un événement, une organisation ou un sujet [12].

9.2.Aperçu sur l'analyse d'opinion

L'analyse d'opinion est une analyse qui permet de s'approfondir sur les avis des clients et de découvrir ce qu'ils aiment et n'aiment pas, et pourquoi afin de crée des produits et services qui répondent à leurs besoins et prendre des décisions stratégiques et tactiques.

9.3.Les différents types d'analyse d'opinion

L'analyse d'opinion peut se diviser aux trois niveaux suivants [13] :

9.3.1. Opinion régulière

Une opinion régulière est le jugement personnel que l'on porte sur un sujet ou un ensemble de sujet qui forcément n'est pas obligatoirement juste.

La figure 5 représente un exemple d'opinion régulière.

Figure 5: Exemple d'opinion régulière.

9.3.2. Opinion comparative

Une opinion comparative est un procédé par lequel on rapproche deux (ou un ensemble) d'entités qui ont au moins un point commun afin de les rapprocher pour examiner leurs différences.

La figure 6 ci-dessous présente un exemple d'opinion comparative.

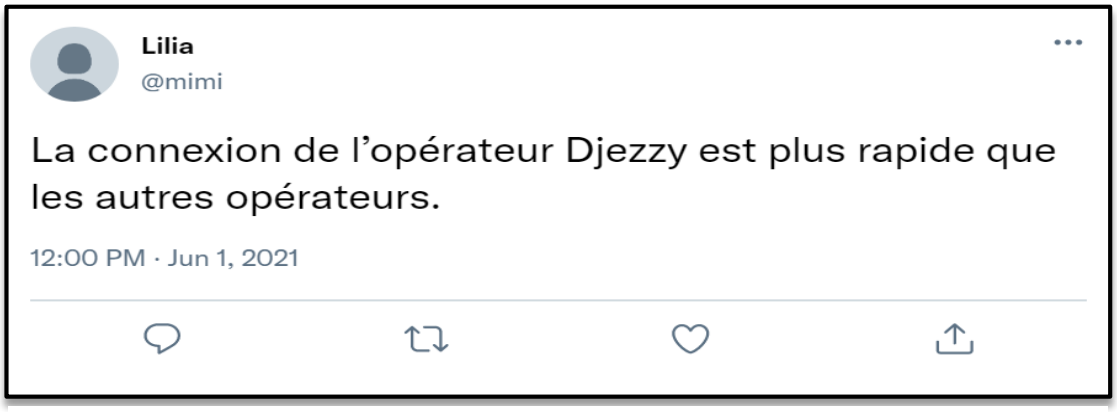


Figure 6 : Exemple d'opinion comparative

9.3.3. Opinion explicite

Une opinion explicite est une expression subjective qui donne une opinion régulière ou comparative, par exemple :

Figure 7: exemple d'opinion explicite

9.3.4. Opinion implicite

Une opinion implicite est une expression objective, qui implique une opinion régulière ou comparative. Et qui exprime généralement un fait souhaitable où Indésirable.

La figure 8 ci-dessous présente un exemple d'opinion comparative.

Figure 8: exemple d'opinion implicite

10. Les approches de l'analyse de sentiment

Plusieurs approches ont été adoptées dans la littérature pour déterminer la polarité d'un texte. Celles-ci peuvent être divisées en trois grandes familles qui sont : les approches basées sur le lexique, les approches basées sur l'apprentissage automatique et les approches hybrides [14].

10.1. Approches basées sur le lexique

Dans cette approche, lors de l'utilisation des techniques de lexique disponibles pour un texte donné, il est question de séparer les mots. En général, cette tâche est effectuée par agrégation de scores : par exemple, les scores de mots subjectifs positifs, négatifs, neutres, etc. sont additionnés séparément pour le même mot. Il attribue une note à chaque mot. Enfin quatre notes sont générées. Celui qui obtient le score maximum donne la répartition globale du texte. Ces approches comportent principalement deux volets :

- Basé sur un dictionnaire.
- Basé sur le corpus.

10.1.1. Approches basées sur un dictionnaire

Dans ce système, l'utilisateur collecte un ensemble de mots de sentiment et établie une liste de départ. Après cela, l'utilisateur commence à rechercher des guides de conversation et un lexique pour trouver des synonymes et des antonymes d'un texte particulier. Une fois cela fait, les substituts nouvellement créés sont ajoutés à la liste de départ jusqu'à ce qu'il n'y ait plus de nouveaux mots trouvés.

10.1.2. Approche basée sur un corpus

Un corpus est essentiellement un terme qui désigne est un groupe ou un ensemble de de documents qui aborde souvent un sujet ou un domaine particulier. En cela, les utilisateurs utilisent l'aide du texte de corpus pour extraire la liste de départ qui est organisée en situation.

L'approche corpus est empirique, analysant les modèles réels d'utilisation de la langue dans les textes naturels. La clé de cette caractéristique de l'Approche Corpus est le langage authentique.

10.2. Approche basée sur l'apprentissage automatique

Les techniques d'apprentissage automatique reposent sur les fameux algorithmes de machine Learning (ML) pour résoudre l'analyse de sentiments en tant que problème de classification de texte régulier qui utilise des fonctionnalités syntaxiques et/ou linguistiques et qui sont coûteuses en termes de temps, d'efforts et de ressources. Les algorithmes de machine Learning sont très souvent utiles pour classer et prédire si un document représente un sentiment positif ou négatif. Les techniques d'apprentissage automatique sont classées en deux types : techniques d'apprentissage traditionnelles et technique d'apprentissage approfondis.

Les techniques traditionnelles sont à leurs tours divisées en deux sous classes, les algorithmes d'apprentissage automatique supervisés et non supervisés [15].

10.2.1. Supervisé

L'apprentissage supervisé a pour but d'établir des règles de comportement à partir d'une base de données contenant des exemples de cas déjà étiquetés. La base de données est en principe un ensemble de couples entrées / sorties {(X, Y)}. Le but est d'apprendre à prédire pour toute nouvelle entrée X, la sortie Y.

Il existe plusieurs algorithmes utilisés pour la classification supervisée tel que :

10.2.2. Non-supervisé

Contrairement à l'apprentissage supervisé, le non supervisé traite le cas où on dispose seulement des entrées {X} sans avoir au préalable les sorties. L'apprentissage non supervisé ou le « clustering » vise à construire des groupes (clusters) d'objets similaires à partir d'un ensemble hétérogène d'objets.

Dans le cadre de notre projet, nous allons particulièrement nous intéresser aux techniques d'Apprentissage Approfondi (Deep Learning en Anglais) et de ce fait, nous allons consacrer un chapitre entier aux modèles et techniques utilisées avant la présentation de la conception de notre solution.

10.3. Approches hybrides

Le concept de méthodes hybrides est très intuitif : il consiste à combiner simplement le meilleur des deux approches, celui basé sur des règles et celui automatique. Généralement, en combinant les deux approches, les nouvelles méthodes peuvent améliorer la précision et la performance en général.

La figure ci-dessous récapitule les différentes approches présentées dans cette section.

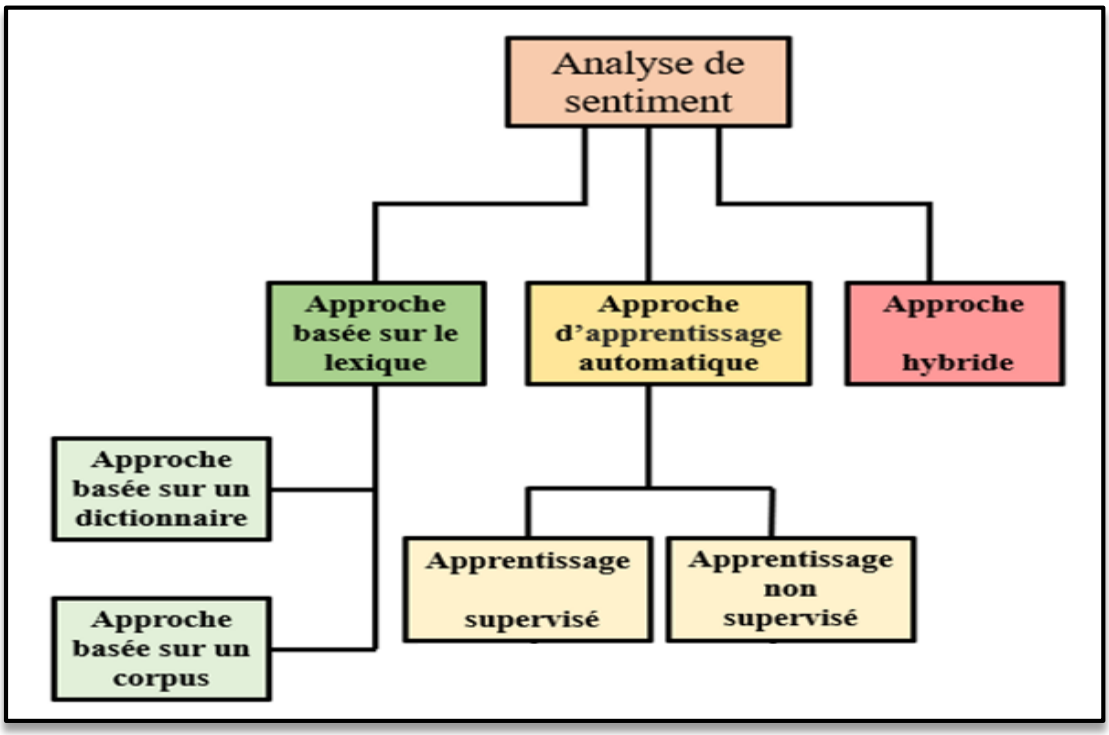


Figure 9 : Diverses approches de l'analyse des sentiments [16]

11. Conclusion

Au cours de ce chapitre, nous avons présenté l'analyse des sentiments, en commençant par ses domaines d'utilisation, ses niveaux et types de classification ainsi

que les approches qui sont adoptées dans la littérature. Nous avons également présenté l'opinion et nous avons montré qu'il existe différentes manières de l'analyser.

Dans le chapitre suivant nous présenterons l'une des langues qui suscite un grand intérêt dans la communauté de l'Analyse des Sentiments et qui est l'**Arabe**, son histoire, ses particularités ainsi que ses dialectes (en mettant l'accent sur le dialecte Algérien).

Chapitre II : La langue Arabe et le dialecte algérien et les notions fondamentales

1. Introduction

Dans le domaine du traitement automatique de la langue, la majorité des recherches menées et des réalisations accomplies ont porté principalement sur l'arabe standard moderne. Les divers dialectes arabes comptent encore parmi les langues sous-dotées. Ce n'est que depuis une dizaine d'années que ces dialectes ont commencé à susciter un intérêt accru au sein de la communauté TAL [17].

Dans ce chapitre, nous allons parler sur la langue arabe, ses défis et ses particularités, puis nous passerons aux différents dialectes arabes nous nous focaliserons sur le dialecte Algérien et pour finir nous allons aborder les notions fondamentales du Deep Learning en nous mettant le point particulièrement sur les architectures des modèles que nous allons utiliser dans notre solution pour l'analyse des sentiments en dialecte algérien.

2. La langue Arabe

La langue arabe est la langue officielle de plus de 22 pays et elle classée comme la 6ème langue la plus parlée dans le monde selon l'Organisation des Nations Unies. Elle est désignée comme étant une langue dont l'histoire et sa culture font partie du patrimoine mondial [18].

La langue arabe appartient à la famille des langues sémitiques, qui se situent géographiquement entre l'Afrique « du Maghreb au Nigéria, le Cameroun, l'Ethiopie, l'Erythrée et la Somalie » et l'Asie « Malte jusqu'à l'Iran » [19].

La carte géographique suivante montre la répartition des langues :

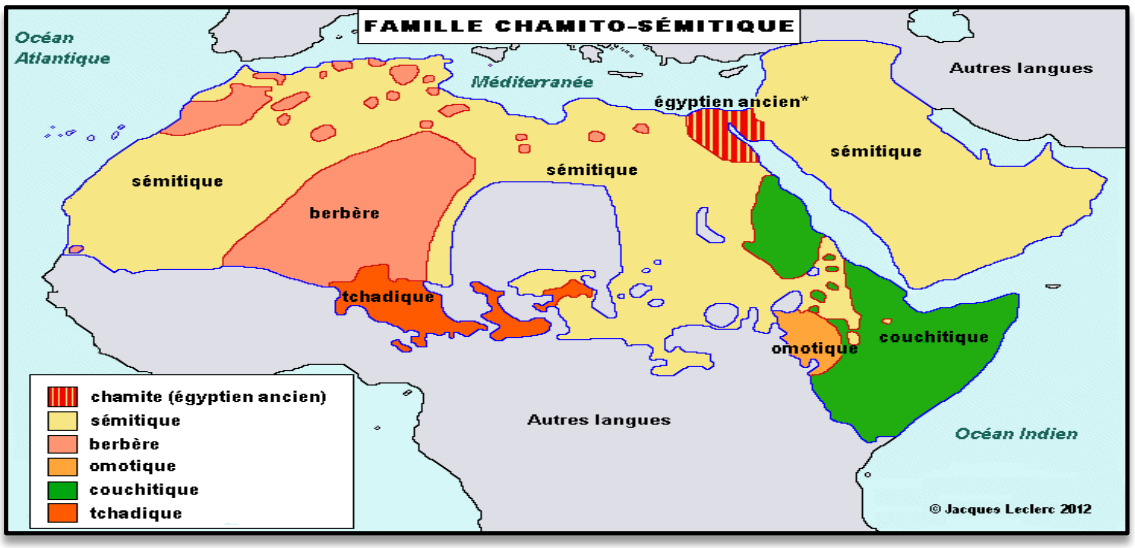


Figure 10:Répartition des langues sémitique [20]

2.1. Les défis de langue arabe

La langue arabe présente aux chercheurs et aux développeurs de traitement automatique du langage naturel (TAL) de sérieux défis [21] :

- L'arabe est à la fois morphologiquement riche et très ambigu.
- En arabe standard moderne (MSA), nous trouvons un ensemble complet de balises [5]de partie du discours.

La langue arabe	La langue anglaise
Compte plus de 300000 balises	Comptes environ 50 balises
Le mot MSA a 12 analyses morphologiques en moyenne	L'anglais à 125000 balises POS[6] par mot en moyenne

Tableau 4: table comparative entre l'arabe et l'anglais

- Le résultat de traitement de l'orthographe arabe omet presque toujours les signes diacritiques utilisés pour spécifier les voyelles courtes et le doublage consonantique.
- L'arabe a des règles morpho-syntaxiques complexes et beaucoup de formes irrégulières : plus de la moitié des pluriels arabes sont irréguliers.
- L'arabe possède un grand nombre de variantes dialectales qui sont aussi différentes du MSA que les langues romanes sont différentes du latin.
- MSA est la forme officielle de l'arabe mais n'est la langue maternelle de personne.
- Les dialectes arabes, véritables langues maternelles, sont majoritairement parlés, n'ont pas de normes écrites et disposent de ressources très limitées.

2.2. Particularité de la langue Arabe

La langue Arabe a certaines particularités qui se résument en [22] :

2.2.1. Absence des voyelles

Dans l'écriture arabe il existe trois types de signes diacritiques : voyelles, nunnation et shadda.

[5] **Balise :** est un processus de conversion d'une phrase en formes liste de mots, liste de tuples (où chaque tuple a une forme (mot, balise))
[6] **POS :** (Part-of-speech) : est un processus qui fait référence à la catégorisation des mots dans un texte en correspondance avec une partie particulière du discours.

2.2.2. Les voyelles

Les voyelles en arabe se composent de trois catégories :

- **Les voyelles brèves (Fatha)**

Cette voyelle est équivalente à la voyelle latine « a », elle se caractérise par un petit trait placé au-dessus de la lettre.

Exemple :

Voyelle brèves (Fatha)	Translitération
بَ	ba
دَ	da
مَ	ma

Tableau 5: Exemple de voyelles brèves fatha

- **Les voyelles brèves (Dama)**

Cette voyelle est équivalente à l'association des deux voyelles *o* et *u* « ou », elle est placée au-dessus de la lettre :

Exemple :

Voyelle brèves (Fatha)	Translitération
مُ	mou
بُ	bou
دُ	dou

Tableau 6 : Exemple de voyelles brèves dama

- **Les voyelles brèves (*kasra*)**

Cette voyelle est équivalente à la voyelle latine « i », elle s'écrit comme la voyelle brèves (fatha) mais cette fois, le trait est placé en dessous du caractère.

Exemple :

Voyelle brèves (Fatha)	Translitération
بِ	bi
دِ	di
مِ	mi

Tableau 7: Exemple de voyelles brèves kasra

2.2.3. Nunation التنوين

C'est une combinaison entre deux voyelles similaires dans les derniers noms, et ce compose de trois formes : **Tanween Damm, Tanween Fath et Tanween Kasr**

Exemple :

Catégorie de nunation	Exemple	Translitération
Tanween dam	مسبحٌ	Masbahoun
Tanween fath	ملعباً	Malabaan
Tanween kasr	نباتٍ	Nabatin

Tableau 8:Exemple de nunation

2.2.4. Shadda الشدة

Elle s'écrit sur la lettre afin de souligner et de prononcer fortement la lettre, la lettre accentuée est la lettre qui indique qu'il y'a deux lettres consécutives, la première d'entre elles est une consonne, et la seconde est mobile.

Elle se compose de trois formes : **Damm, Fath et Kasr.**

Exemple :

Catégorie de nunation	Exemple	Translitération
dam	عَلَّمَ	Allama
fath	يُصَوِّرُ	Yousawwiro
kasr	مُعَلِّم	Mouaalim

Tableau 9: Exemple de voyelles brèves shadda

A la fin de cette partie on peut déduire que l'écriture arabe peut être totalement voyellé (présence les voyelles brèves, shadda, nunation) et même partiellement voyellé et cela est dû au fait que les mots non voyellé possèdent plusieurs voyellations possibles.

2.3. Les dialectes arabes

Avec au moins 350 millions de locuteurs, l'arabe est la cinquième langue la plus parlée au monde après l'anglais, le mandarin, l'hindi, l'espagnol et le français.

Cependant, le type d'arabe parlé du Moyen-Orient et d'Afrique du Nord peut différer considérablement d'un pays à l'autre.

Les différences entre le dialecte maghrébin en Afrique du Nord, par exemple, et le dialecte irakien (mésopotamien) peuvent être si importantes que les locuteurs natifs de chacun peuvent ne pas être capables de distinguer plus de quelques mots.

2.3.1. L'historique des dialectes arabes

Tous les dialectes arabes parlés aujourd'hui, que ce soit en Afrique du Nord ou dans le Golfe, ont une origine commune dans les formes d'arabe parlées dans la péninsule arabique autour de la formation de l'islam au début du VIIe siècle. Ces dialectes se sont répandus avec l'avancée des armées musulmanes vers de nouvelles terres à travers le Moyen-Orient et l'Afrique du Nord et se sont ensuite développés dans les formes que nous connaissons aujourd'hui. [22].

2.3.2. Différentes variétés arabes dans le monde arabe

Les principaux groupes de variantes comprennent [23] :

- **Maghrebi (parfois appelé arabe occidental)**

Généralement censé inclure le Maroc, l'Algérie, la Tunisie, la Libye, le Sahara occidental et la Mauritanie, ces dialectes sont appelés Derja, Derija ou Darija par les personnes qui les utilisent.

Ce dialecte a été et continue d'être influencé par d'autres langues. Il s'agit notamment du latin africain éteint, du vieil arabe, du français, du mozarabe, de l'italien, de l'espagnol, du turc et des langues du Niger-Congo.

- **Soudanais**

Ce sont les variantes de la langue arabe parlées au Soudan et en Érythrée. Elles sont étroitement liées à l'arabe égyptien et contiennent des influences nubiennes tout en conservant certaines formes de prononciation archaïques.

- **Égyptien**

L'égyptien copte était la langue prédominante de l'Égypte avant la conquête musulmane de la région au 7^{e} siècle. L'arabe égyptien d'aujourd'hui, parfois simplement appelé égyptien en est fortement influencé.

- **Arabe du sud (Péninsule arabique)**

La péninsule arabique est le berceau de la langue arabe. Les dialectes parlés dans cette région sont les plus proches de l'arabe classique de la littérature ancienne et il existe un grand nombre de ce que les locuteurs d'autres régions pourraient appeler des caractéristiques plus archaïques.

Les pays dans lesquels on trouve des locuteurs de l'arabe péninsulaire sont le Bahreïn, le Koweït, Oman, le Qatar, l'Arabie saoudite, les Émirats arabes unis, le Yémen, le sud de l'Irak et les peuples tribaux de la Jordanie.

- **Mésopotamien**

Il est parlé dans certaines parties de l'Irak, de la Syrie, de l'Iran et du sud-est de la Turquie.

- **Arabe levantin**

L'arabe levantin est parlé le long de la côte de la mer Levantine dans des pays comme la Jordanie, le Liban, la Palestine, la Syrie et la Turquie.

La figure ci-dessous (figure 07) illustre quelques exemples de traduction de la phrase « Qu'est-ce qu'il y a de nouveau ? » en Anglais « what's up new ? dans quelques dialectes Arabes.

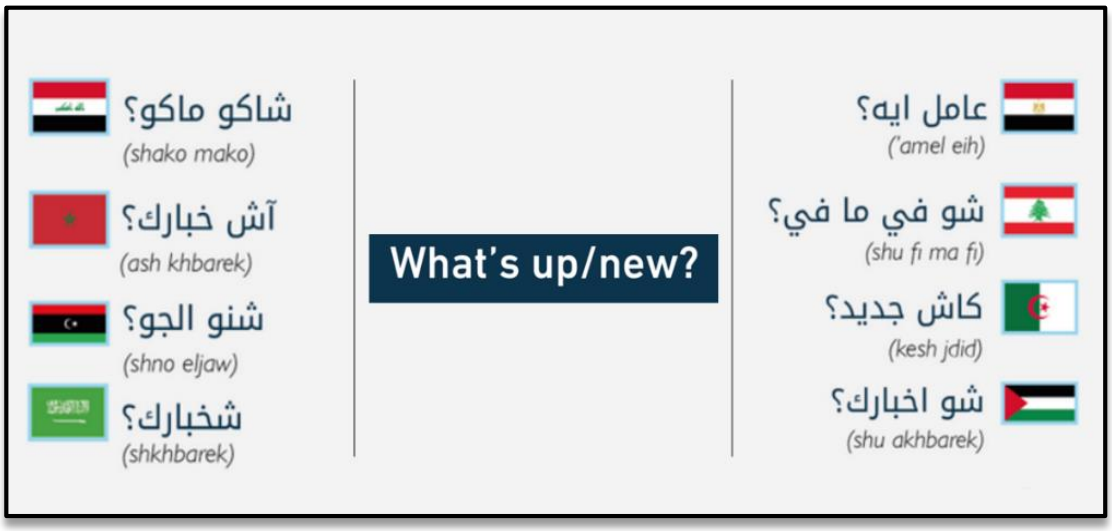


Figure 11: Exemple de quelques dialectes arabes

3. Le dialecte Algérien

Le Dialecte Algérien est enrichi par les langues des pays ayant colonisé ou géré la population algérienne au cours de l'histoire du pays. Parmi les langues de ces groupes, citons le Turc, l'Espagnol, l'Italien et plus récemment le Français. De ce fait, au sein du dialecte algérien, nous trouvons des mots tels que « سلام» Salam 'saluer' et ayant comme origine l'arabe standard moderne, « جرنان » Jernan 'Journal' étant originaire du français, « بلاك» Balak 'Peut-être' issu du turc, « دادا » Dadda 'Monsieur' emprunté du berbère.

Aussi, le dialecte Algérien se caractérise par quatre grandes variétés régionales : L'Algérois, qui couvre toute la zone centrale du pays, l'Oranais à l'Ouest, le Chaoui et le kabyle à l'Est, le berbère ainsi que ses variétés au Sud du pays. [24]

3.1. Complexité du dialecte algérien

Le dialecte algérien est caractérisé par l'absence de standard et de ressources. Il diffère de l'arabe classique en termes de représentation linguistique (phonologique et morphologique), et de lexique ainsi que dans la représentation syntaxique [25].

Parmi les caractéristiques du dialecte algérien [26] :

- Langage informel : pas de règles strictes en écriture.
- La négation est exprimée par des mots séparés contrairement à l'arabe.

Exemple : 'لم يعجبني' se traduit en dialecte algérien : 'ماعجبنيش'.

- Mots de plusieurs langues dans un même texte.
- Confusion d'interprétation d'un mot entre l'Arabe standard et le dialecte Algérien, un mot peut avoir des sens différents.
- Ecriture en lettres arabes ou latines (l'arabizi).

  Exemple : 3jebni 'عجبني'.
- Ecriture sous forme de langage SMS.

  Exemple : « hmd » en lettres Arabizi qui signifie «الحمد لله » « Grâce à Dieu ».
- Un seul terme Algérien peut dépasser 10 formes différentes en Arabizi.

  Exemple : « 3jebni », « 3ajebni », « ajebni ».
- Utilisation des lettres latines et certains chiffres pour désigner des sons qui n'existent pas en français.

  Exemple : « 3 » pour la lettre « ع» pour éviter d'écrire « a » qui peut être confondu avec « أ» et « 7 » pour « ح» pour éviter d'écrire "H" qui peut être confondu avec « هـ»
- Plusieurs mots peuvent avoir un seul sens.

  Exemple : 'انعم', 'وي','واه',' ايه' qui signifient « نعم » « Oui ».

Le dialecte Algérien a aussi des spécificités, tel que la conjugaison, la négation, la signification des mots.

3.2. La conjugaison pour le dialecte algérien

La conjugaison pour le dialecte algérien inclut l'ajout d'un ensemble de préfixes et de suffixes à un lemme donné et varient selon le pronom utilisé.

Exemple :

Verbe	Présent	Traduction	Passé composé	Traduction
كتب Ecrire	نكتب	j'écris	كتبت	j'ai écrit
	تكتب	tu écris	كتبت	tu as écrit
	تكتب /يكتب	il/elle écrit	كتبت /كتب	Il / elle a écrit
	نكتبو	nous écrivons	كتبنا	nous avons écrit
	تكتبو	vous écrivez	كتبتو	vous avez écrit
	يكتبو	ils/elles écrivent	كتبو	ils ont écrit

Tableau 10: **Conjugaison du verbe 'Ecrire' au présent et passé composé**

À partir du Tableau 10, nous remarquons que les lettres ي, ت, ن représentent les préfixes et les lettres ت, و, بو, ي, نا ,توreprésentent les suffixes pour les termes en dialecte Algérien.

3.2.1. La négation pour le dialecte Algérien

La négation pour les termes en dialecte Algérien peut se faire de deux manières, selon les cas d'utilisation :

- à l'aide des lettres : ما, ش et à l'aide du mot : ماشي

Exemple :

Phrase	Négation de la phrase
عرض مليح	عرض ماشي مليح
تحيا جازي كونفور	ماتحياش جازي كونفور

Tableau 11: La négation pour le dialecte Algérien.

A Partir du tableau 12, nous remarquons que pour rendre la négation on met 'ما' devant le mot sur lequel porte la négation. On ajoute aussi dans certains cas après ce mot le Substantif 'ش' qui joue le même rôle que « pas » en français.

3.2.2. La signification des mots

Il est possible d'identifier différentes significations associées à un mot pour la simple et bonne raison qu'un mot peut avoir plusieurs sens dans différents contextes.

Phrase	Signification de la phrase
هاذ العرض واعر	Cette offre est excellente
هاذ الامتحان واعر	Cet examen est difficile

Tableau 12: Exemple de différentes significations d'un mot

3.3. L'Arabizi

En Arabizi les textes sont rédigés en caractères latins. De plus, les lettres arabes qui n'ont pas d'équivalent en latin sont remplacées par des chiffres.

Exemple :

Lettre arabe	Transcription	Chiffre
ع	a	3
ق	ka	9
ح	ha	7

Tableau 13: Exemple de lettres arabe et leurs équivalents en lettre et transcription

La distinction entre les voyelles disparaît dans ce type d'écriture vu l'usage d'un seul alphabet, mais il permet l'usage correcte de l'arabe dialectal.

4. Apprentissage en profondeur

L'apprentissage en profondeur (de son appellation Anglaise : Deep Learning) est un sous-domaine de l'apprentissage automatique concerné par des algorithmes inspirés de la structure et de la fonction du cerveau humains appelés Réseaux de Neurones Artificiels en référence aux réseaux de neurones. Les architectures d'apprentissage en profondeur offrent d'énormes avantages pour la classification de texte par rapport aux modèles standards. Alors que les algorithmes traditionnels sont linéaires, les modèles d'apprentissage en profondeur, généralement les réseaux de neurones, sont empilés dans une hiérarchie de complexité et d'abstraction croissantes (d'où le « profond » dans l'apprentissage en profondeur). [27]

Le schéma général d'une application d'analyse de sentiments basée sur les réseaux de neurones en profondeur est résumé dans la figure suivante (figure 15)

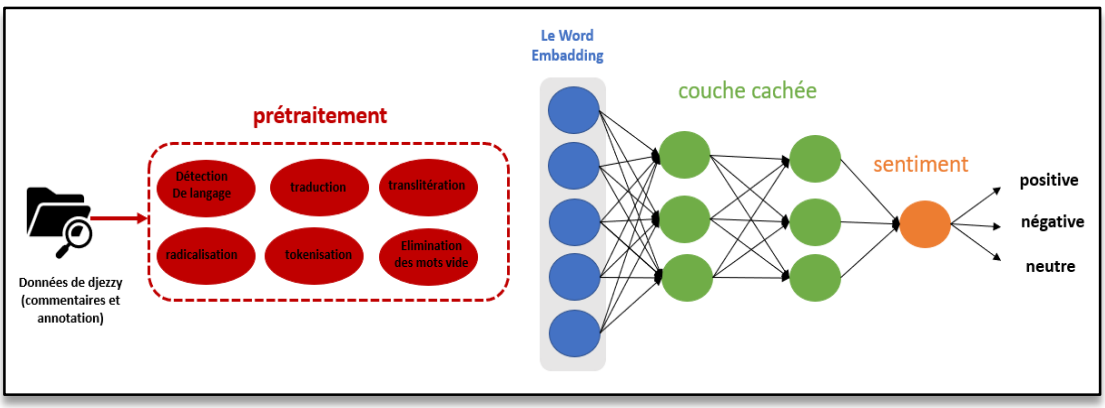


Figure 12: Analyse des sentiments basée sur le Deep Learning [28]

Il existe essentiellement 2 approches pour l'apprentissage en profondeur [29] :

4.1. Apprentissage profond classique

Certes, construire un modèle de réseau à partir de zéro sans l'influence d'autres recherches antérieures ou de modèles précédemment construits n'est pas chose aisée. Il faut faire tous les processus de développement de modèles depuis le début comme :

- **Nettoyage/imputation/prétraitement des données :** Le nettoyage des données est le processus de préparation des données pour l'analyse en supprimant ou en modifiant les données incorrectes, incomplètes, non pertinentes, dupliquées ou mal formatées.
- **Extraction de caractéristiques :** L'extraction de caractéristiques fait référence au processus de transformation des données brutes en caractéristiques numériques qui peuvent être prises en charge tout en préservant les informations de l'ensemble de données d'origine. Cela donne de meilleurs résultats que d'appliquer l'apprentissage automatique directement aux données brutes.
- **Création de Réseau de neurones artificiels :** Il existe deux bibliothèques principales pour construire des réseaux de neurones : TensorFlow (développé par Google) et PyTorch (développé par Facebook). Ils peuvent effectuer des tâches similaires, mais le premier est plus adapté pour la production tandis que le second est bon pour la construction de prototypes rapides car il est plus facile dans la phase d'apprentissage.

La figure qui suit (figure 12) illustre les constituants d'un réseau de neurones artificiel.

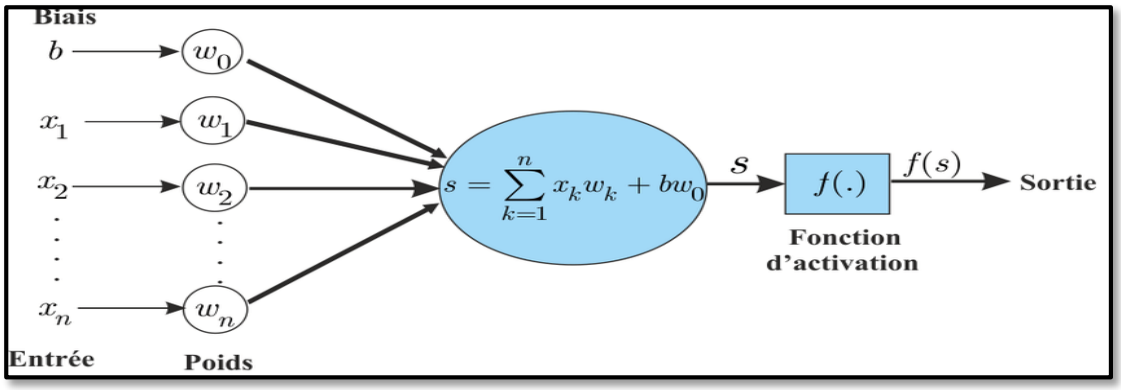


Figure 13: Réseau de neurones artificiels [30]

Les réseaux de neurones artificiels sont constitués de couches qui peuvent être comme suit :

❖ **Couche d'entrée :** pour tâche de transmettre le vecteur d'entrée au réseau de neurones.

❖ **Les couches cachées :** représentent les nœuds intermédiaires, elles appliquent plusieurs transformations aux nombres afin d'améliorer la précision du résultat final, et la sortie est définie par le nombre de neurones.

❖ **Couche de sortie :** qui renvoie la sortie finale du réseau de neurones. Dans le cas d'une classification multi classe avec 3 classes différentes, la couche de sortie devra comporter 3 neurones.

Avec une dimension d'entrée et une dimension de sortie. Ce dernier est déterminé par le nombre de **neurones**, une unité de calcul qui relie les entrées pondérées via une **fonction d'activation** (qui aide le neurone à s'activer). Les **poids**, comme dans la plupart des algorithmes d'apprentissage automatique, sont initialisés de manière aléatoire et optimisés lors de l'apprentissage afin de minimiser une fonction de perte.

Remarque : On pourrait dire que tous les modèles d'apprentissage en profondeur sont des réseaux de neurones, mais tous les réseaux de neurones ne sont pas des modèles d'apprentissage en profondeur.

De manière générale, le "Deep" Learning s'applique lorsque l'algorithme possède au moins 2 couches. Les notions suivantes sont à prendre en considération dans tout réseau de neurones [31] :

- **Initialisation du poids :** L'initialisation des poids est une procédure permettant de définir les poids d'un réseau de neurones sur de petites valeurs aléatoires qui définissent le point de départ de l'optimisation (apprentissage ou formation) du modèle de réseau de neurones.
- **La régularisation :** Un ensemble de techniques qui peuvent empêcher le surajustement dans les réseaux de neurones et ainsi améliorer la précision d'un modèle d'apprentissage en profondeur face à des données complètement nouvelles du domaine problématique.

La figure ci-dessous représente un modèle d'apprentissage avec un bon ajustement et une variance élevée.

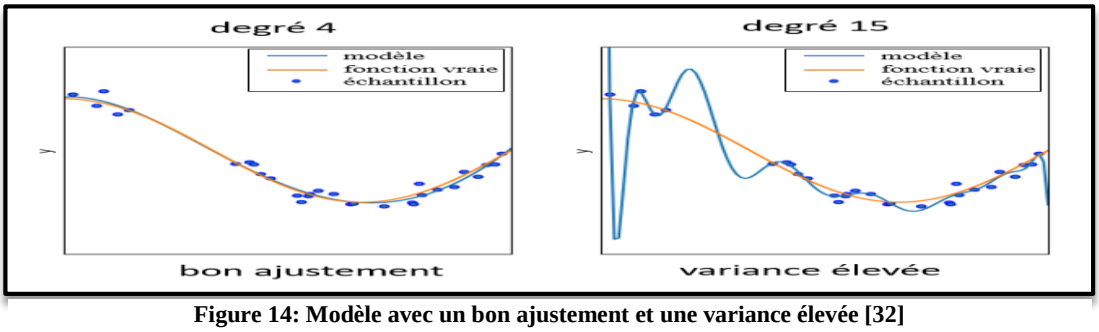


Figure 14: Modèle avec un bon ajustement et une variance élevée [32]

- **réglage des hyperparamètres** consiste à trouver un ensemble de valeurs d'hyperparamètres optimales pour un algorithme d'apprentissage tout en appliquant cet algorithme optimisé à n'importe quel ensemble de données. Cette combinaison d'hyperparamètres maximise les performances du modèle et minimisant une fonction de perte prédéfinie pour produire de meilleurs résultats avec moins d'erreurs.

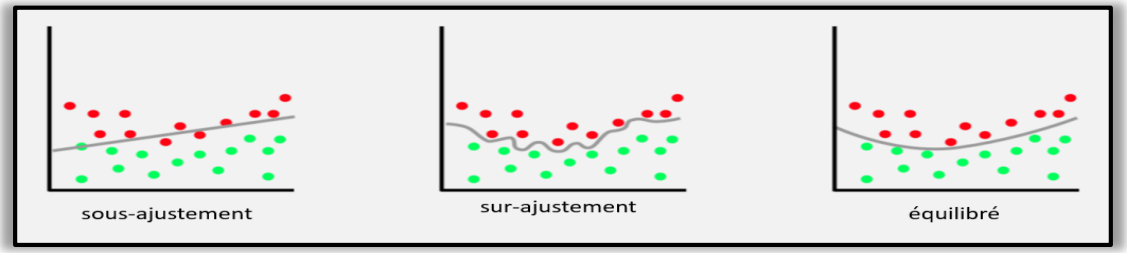


Figure 15: sous Ajustement et Sous-ajustement et réajustement du modèle [33]

- **Étudier les inférences du modèle :** L'inférence applique les connaissances d'un modèle de réseau neuronal formé et les utiliser pour déduire un résultat. Ainsi, lorsqu'un nouvel ensemble de données inconnu est entré via un réseau neuronal formé, il produit une prédiction basée sur la précision prédictive du réseau neuronal.

La figure ci-dessous représente quelque modèle se basant sur l'apprentissage profond classique :

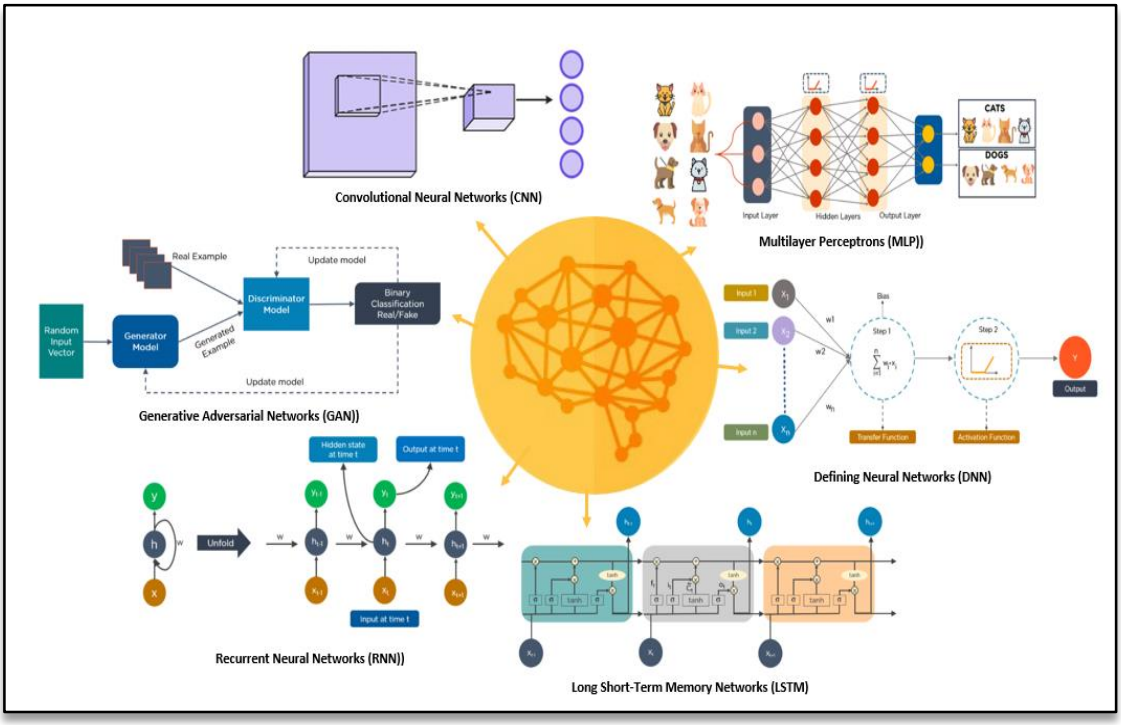


Figure 16: modèles basant sur l'apprentissage profond classique

4.1.1. Architecture des modèles utilisés

4.1.1.1. Le modèle CNN

Les réseaux de neurones convolutionnels (de l'Anglais Convolutional Neural Networks CNN) sont construits pour une analyse fine des données exprimées sous la forme d'une

grille, typiquement les pixels d'une image. Ils ont montré leur efficacité sur l'analyse de textes ou le traitement d'images. Un réseau CNN est construit par la succession d'une couche de convolution et d'une couche d'agrégation de l'information (Pooling) et se termine par une couche de neurones totalement connectés.

Sur la figure ci-dessous, on peut retrouver l'architecture d'un réseau CNN pour une tâche de classification de textes :

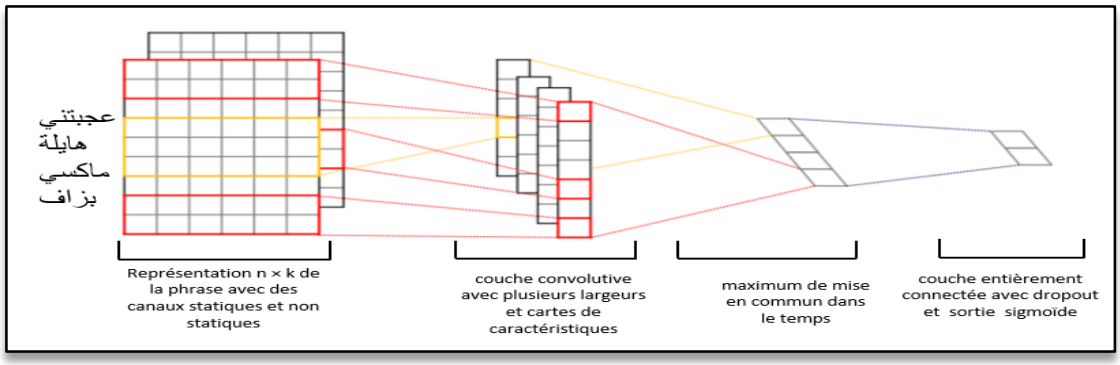


Figure 17: Représentation de la structure d'un réseau CNN pour une analyse d'un commentaire

Lorsque les CNN sont appliqués à du texte au lieu d'images, nous avons un tableau à 1 dimension représentant le texte. Ici, l'architecture des CNN est remplacée par des opérations de convolution et de mise en commun 1D.

Remarque : la classification d'une phrase dans un ensemble de catégories prédéterminées en considérant les n-grammes, s'agit de mots ou d'une séquence de mots, ou également de caractères ou d'une séquence de caractères.

Les principaux paramètres des réseaux de neurones cnn pour la classification des textes sont :

- **Convolutions 1-D sur le texte**

Représente une couche qui peut être utilisée pour détecter des caractéristiques dans un vecteur [34]**.**

Soit la séquence de mots $w_{1:n} = w_1, \ldots, w_n$ suivante :

Où chacun est associé à un vecteur plongeant de dimension ***d***. Une convolution 1D de largeur ***-k*** est le résultat du déplacement d'une fenêtre glissante de taille ***k*** sur la phrase et de l'application du même filtre de convolution à chaque fenêtre de la séquence, c'est-à-dire un produit scalaire entre la $w_i, \ldots, w_{i+k}$ concaténation des vecteurs d'intégration dans une fenêtre donnée et un vecteur de poids ***u***, qui est alors souvent suivi d'une fonction d'activation non linéaire ***g***.

En considérant une fenêtre de mots, le vecteur concaténé de la ième fenêtre est alors :

$$x_i = [w_i, w_{i+1}, \ldots, w_{i+k}] \in R^{k \times d}$$

Le filtre de convolution est appliqué à chaque fenêtre, ce qui donne des valeurs scalaires, chacune pour la i éme $r_i = g(x_i \cdot u) \in R$ fenêtre :
En pratique on applique typiquement plusieurs filtres, qui $u_1, \ldots, u_l$ peuvent alors être représentés comme un vecteur multiplié par une matrice $\boldsymbol{u}$ et avec l'ajout d'un terme de biais $\boldsymbol{b}$:

$$\mathrm{r}_i = g(x_i \cdot U + b)$$

Avec

$$\mathrm{r}_i \in R^l, \quad x_i \in R^{k \times d}, \quad U \in R^{k \cdot d \times l} \text{ and } \quad b \in R^l$$

- **Kernel**

Une matrice modèle qui est utilisée dans l'opération de convolution.

- **Pooling**

L'opération de mise en commun est utilisée pour combiner les vecteurs résultant de différentes fenêtres de convolution en un vecteur unidimensionnel. Ceci est refait en prenant le max ou la valeur moyenne observée dans le vecteur résultant des convolutions. Idéalement, ce vecteur capturera les caractéristiques les plus pertinentes de la phrase/du document.
Ce vecteur est ensuite alimenté plus bas dans le réseau pour effectuer une prédiction.

- **Filtres**

Le nombre des filtres convolutionnels à inclure dans la couche.

- **feature_maps**

Le nombre de cartes d'entités qui contrôlent directement la capacité et dépendent du nombre d'exemples disponibles et de la complexité des tâches.

- **Pudding**

Same entraîne un remplissage de l'entrée de sorte que la sortie ait la même longueur que l'entrée d'origine. ***Valide*** signifie "pas de remplissage".

4.1.1.2. Le modèle LSTM

Long Short-Terme Memory Network ou LSTM, est une variante d'un réseau neuronal récurrent (RNN) qui est assez efficace pour prédire les longues séquences de données comme les phrases et les cours des actions sur une période.

Il diffère d'un réseau à anticipation normal car il existe une boucle de rétroaction dans son architecture. Il comprend également une unité spéciale connue sous le nom de cellule de mémoire pour retenir les informations passées plus longtemps pour faire une prédiction efficace [35].

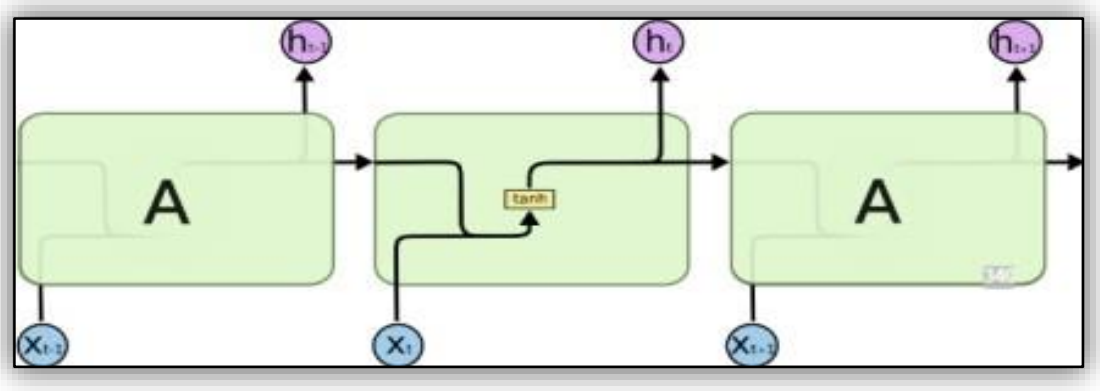


Figure 18 : cellules de RNN

En fait, LSTM avec ses cellules mémoire est une version améliorée des RNN traditionnels qui ne peuvent pas prédire en utilisant une si longue séquence de données et se heurtent au problème du gradient de fuit sur la figure ci-dessous :

- ❖ Le mot vecteur est la sortie du bloc à la suivante en entrée.
- ❖ Les rectangles jaunes sont une couche de réseaux de neurones appris.
- ❖ Les lignes fusionnant les unes avec les autres effectuent une concaténation.
- ❖ Le contenu est copié et va à différents endroits avec la ligne de branchement où Xt est l'entrée à l'étape t et HT est la sortie à l'instant t.

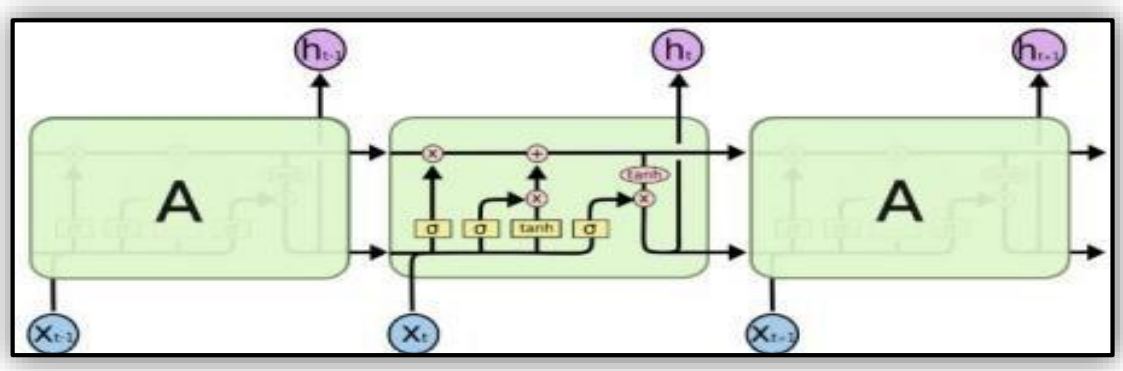
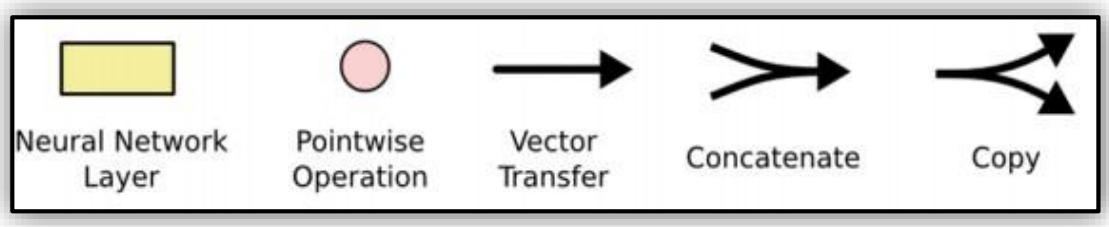


Figure 19: cellules de LSTM



- **Description de la cellule LSTM**

Une cellule LSTM agit comme convoyeur d'information modulé à l'aide de 3 portes configurables.

- **Porte d'oubli**

Décide quoi garder du contexte précédent pour mettre à jour l'état de la cellule, basé sur l'entrée courante et sa formule est $f_t = \sigma\ (W_f.[h_{t-1}, x_t] + b_f)$ avec :

❖ ht−1 : La sortie a l'instant t-1

❖ xt : L'entrée courant à l' instant t

❖ b_f : le bais

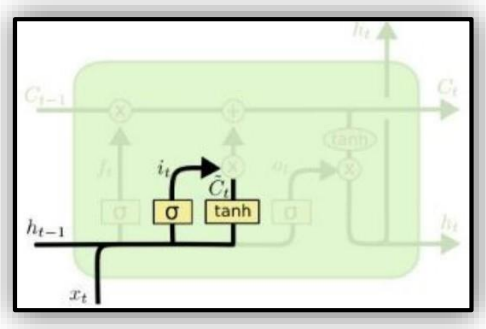
❖ w_f : le poids

❖ σ : C'est la fonction sigmoïde

- **porte d'entré**

Qui décide quoi contribuer de la mémoire (**$\tilde{C}$**t est une valeur candidate est équivalent à l'état caché) pour mettre à jour l'état de la cellule, basé sur Xt.

tanh : C'est la fonction d'activation tangente hyperbolique.



$$i_t = \sigma\ (W_i.[h_{t-1}, x_t] + b_i)$$

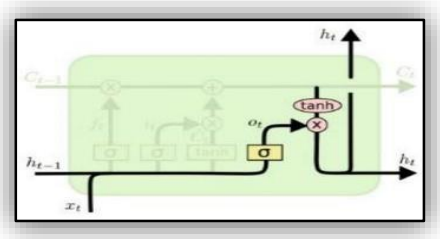
$$\check{C}_t = tanh\ (W_C.[h_{t-1}, x_t] + b_C)$$

- **Mise à jour de la cellule**

On combine ce qui a été retenu de l'état précédent de la cellule avec celui retenu de la mémoire comme suit : $C_t = f_t * C_{t-1} + i_t * \check{C}_t$ avec Ct est l'état interne.

- **Porte de sortie**

Décide quoi rendre publique du nouvel état de la cellule.



$$o_t = \sigma\,(W_0[h_{t-1}, x_t] + b_0)$$

$$h_t = o_t * tanh(C_t)$$

Avec :h_t : La sortie

La figure ci-dessous montre l'architecture globale d'une seule cellule de Lstm :

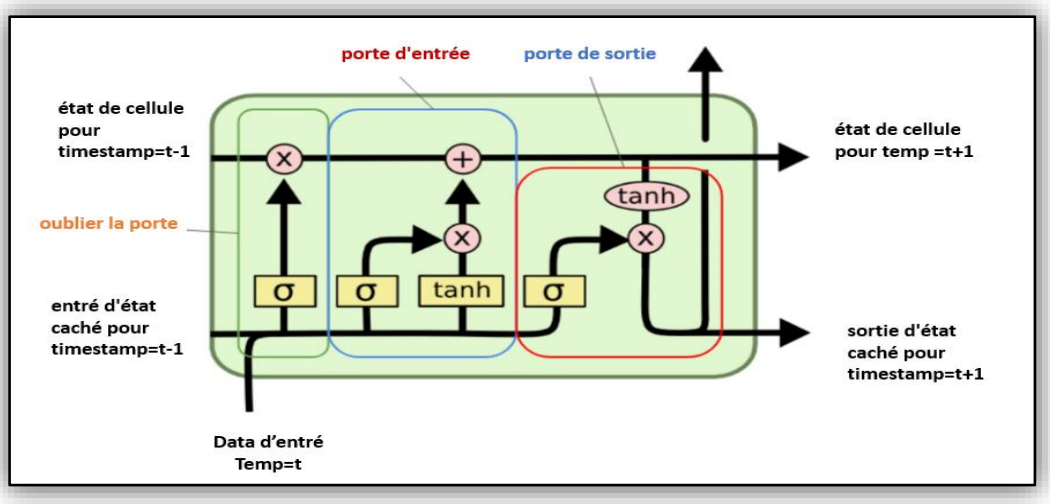


Figure 20 : l'architecture d'une seule cellule LSTM

4.1.1.3. Le modèle bi-lstm

Les LSTM bidirectionnels sont une extension des LSTM traditionnels qui peuvent améliorer les performances du modèle sur les problèmes de classification de séquences. Dans les problèmes où tous les pas de temps de la séquence d'entrée sont disponibles, les LSTM bidirectionnels entraînent deux LSTM au lieu d'un sur la séquence d'entrée avec cette formule mathématique :

$$\hat{y}<t> = g(Wy[\vec{a}<t>, \overleftarrow{a}<t>] + by)$$

$\vec{a}^{<t>}$ est la sortie de Lstm de gauche à droite et $\overleftarrow{a}^{<t>}$ est la sortie de Lstm inverse.

La figure ci-dessous présente l'architecture globale du Lstm bidirectionnelle :

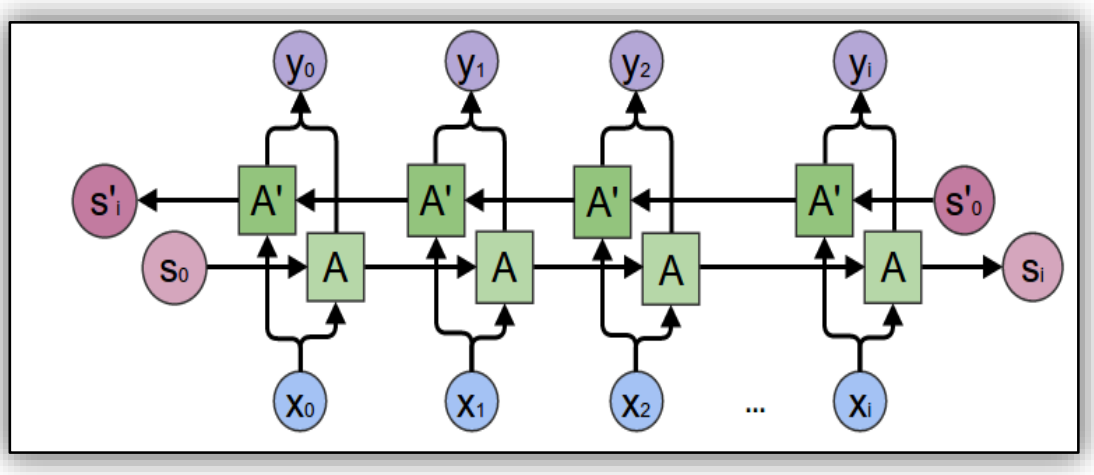


Figure 21: l'architecture du LSTM bidirectionnelle

4.1.1.4. Le modèle CNN-LSTM

Nous pouvons définir un modèle CNN-LSTM à former conjointement dans la bibliothèque Keras.

Un CNN-LSTM peut être défini en ajoutant des couches CNN sur le frontend, suivies de couches LSTM avec une couche Dense sur la sortie.

Il est utile de considérer cette architecture comme définissant deux sous-modèles : le modèle CNN pour l'extraction de caractéristiques et le modèle LSTM pour l'interprétation des caractéristiques à travers les pas de temps.

La figure ci-dessous présente l'architecture du modèle cnn-lstm :

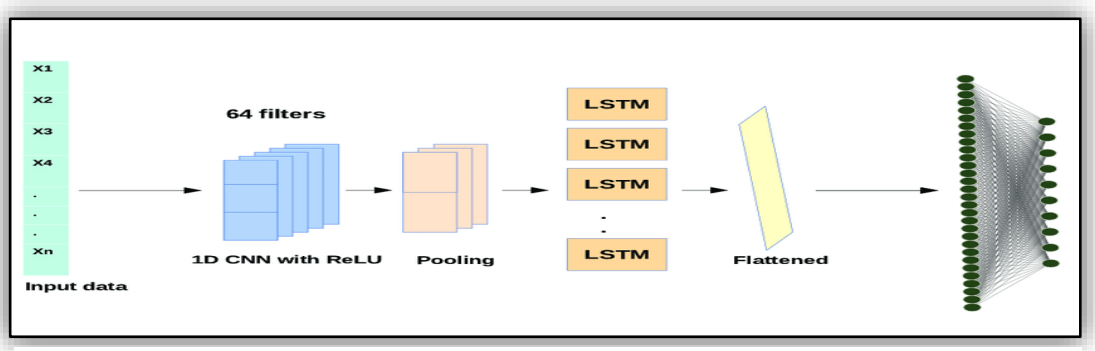


Figure 22 : l'architecture du CNN-LSTM [36]

4.2. Apprentissage par transfert

L'apprentissage par transfert est la formation d'un modèle de réseau sur l'ensemble de données donné et l'utilisation de ce modèle pour d'autres ensembles de données avec les mêmes caractéristiques et une distribution différente des classes (parfois avec des classes différentes).

Il n'y a pas de règles strictes selon lesquelles tout dans le deuxième ensemble de données doit être identique au premier en termes de fonctionnalités voire d'autres

paramètres. Des améliorations de bas niveau sont recommandées et peuvent être effectuées en fonction du caractère du nouvel ensemble de données et d'un nouveau problème.

La figure ci-dessous représente quelque modèle se basant sur l'apprentissage par transfert :

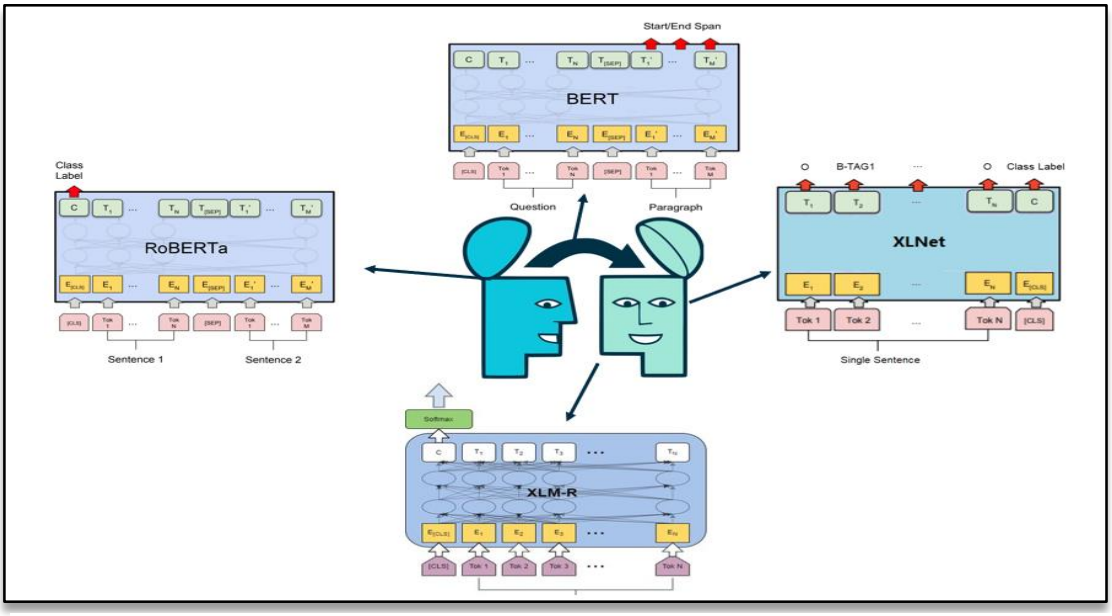


Figure 23: Modèles se basant sur l'apprentissage par transfert

Remarque : dans notre étude nous nous sommes focalisés sur un seul modèle qui se base sur l'apprentissage par transfert. Il s'agit de l'un des modèles les populaire qui se base sur l'apprentissage par transfert qui est Bert.

4.2.1. Bert

Nous introduisons dans notre solution un modèle de représentation du langage appelé BERT, qui signifie en anglais « Bidirectional Encoder Representations from Transformers », ce modèle utilise le transformeur, un mécanisme d'attention qui gère les relations des contextes entre les mots dans un texte. Contrairement aux modèles directionnels, qui lisent le texte de manière séquentielle (de gauche à droite ou de droite à gauche) ce modèle lit la séquence entière des mots dans un texte une seule fois.

Le transformeur comprend deux mécanismes : un encodeur qui lit l'entrée du texte et un décodeur qui produit la prédiction [37].

- **Le fonctionnement de Bert**

BERT utilise Transformer, un mécanisme d'attention qui apprend les relations contextuelles entre les mots (ou sous-mots) dans un texte. Dans sa forme vanille,

Transformer comprend deux mécanismes distincts - un encodeur qui lit l'entrée du texte et un décodeur qui produit une prédiction pour la tâche. Puisque l'objectif de BERT est de générer un modèle de langage, seul le mécanisme d'encodeur est nécessaire.

L'encodeur Transformer lit la séquence entière de mots en une seule fois. Par conséquent, il est considéré comme bidirectionnel, bien qu'il soit plus exact de dire qu'il est non directionnel. Cette caractéristique permet au modèle d'apprendre le contexte d'un mot en fonction de l'ensemble de son environnement (gauche et droite du mot).

L'entrée est une séquence de jetons, qui sont d'abord intégrés dans des vecteurs, puis traités dans le réseau neuronal. La sortie est une séquence de vecteurs de taille H, dans laquelle chaque vecteur correspond à un jeton d'entrée avec le même index.

Une approche directionnelle qui limite intrinsèquement l'apprentissage du contexte. Pour surmonter le défi de définir un objectif de prédiction, BERT utilise deux stratégies de formation :

❖ **MLM**

Le modèle de langage masqué (MLM) est le processus par lequel le BERT a été préformé. Il a été démontré que l'application du MLM sur ses propres données peut améliorer les performances.

❖ **NSP**

Dans le processus de formation BERT, le modèle reçoit des paires de phrases en entrée et apprend à prédire si la deuxième phrase de la paire est la phrase suivante dans le document d'origine.

La figure ci-dessous représente les entées du modèle BERT :

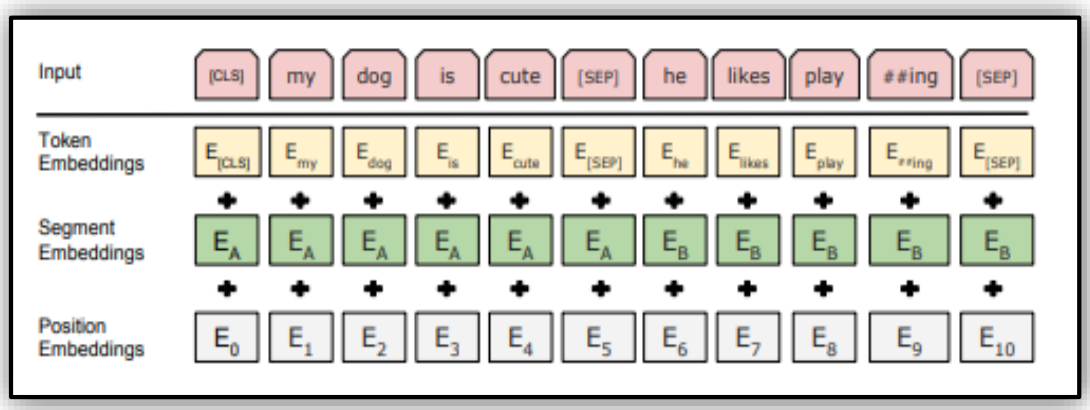


Figure 24: Représentation des entrées du modèle BERT.

4.3. Le Word Embedding

Le Word Embedding est une représentation vectorielle d'un texte arrangé par similarité de mots. Ce type de représentation aide à présenter les informations dans des vecteurs de dimension inférieure et à extraire la signification sémantique des mots en les mappant dans un espace géométrique. Cette approche de la représentation des mots et des documents qui peut être considérée comme l'une des principales percées de l'apprentissage en profondeur sur les problèmes complexes de traitement du langage naturel. [38]

4.4. Métriques d'évaluation

L'évaluation de l'apprentissage des modèles est une partie essentielle de tout le projet. Le modèle peut donner des résultats satisfaisants lorsqu'il est évalué à l'aide des métriques.

Il existe plusieurs façons d'évaluer les modèles de Deep Learning. Voici quelques exemples [39] :

4.4.1. Val_Loss, Val_Accuracy

Lorsque nous formons le modèle en keras, la précision et la perte dans le modèle keras pour les données de validation peuvent varier selon les cas. Habituellement, à chaque époque croissante, la perte devrait diminuer et la précision devrait augmenter. Mais avec **Val_Loss** (perte de validation keras) et **Val_ACC** (précision de validation keras), de nombreux cas peuvent être possibles comme ci-dessous :

- **Val_Loss** commence à augmenter, **Val_ACC** commence à diminuer. Cela signifie que le modèle accumule des valeurs et n'apprend pas
- **Val_Loss** commence à augmenter, **Val_ACC** augmente également. Cela peut être un cas de surajustement ou de valeurs de probabilité diverses dans les cas où Softmax est utilisé dans la couche de sortie
- **Val_Loss** commence à diminuer, **Val_ACC** commence à augmenter. C'est également très bien car cela signifie que le modèle construit apprend et fonctionne bien.

4.4.2. Précision, rappel, Accuracy, F1 score

Avant d'expliquer ces métriques, commençons par définir les 4 termes importants à leur calcul.

❖ **Vrais positifs :** Les cas dans lesquels la prédiction de l'humain était **positive** et celle de la machine ou du système (sortie réelle) était également positive (positif réel ou True positif de son appellation en Anglais True Positif).

❖ **Vrais négatifs :** Les cas dans lesquels la prédiction de l'humain était ***négative*** et la sortie réelle était ***négative*** aussi (True negatif en Anglais).

❖ **Faux positifs :** Les cas dans lesquels l'humain a prédit négatif (non) et la sortie réelle était positive (oui).

❖ **Faux négatifs :** Les cas dans lesquels la prédiction de l'humain était ***négative*** et la sortie réelle était positive (en Anglais False negatif).

- **Précision**

Cette une métrique qui calcule le nombre de résultats positifs corrects divisé par le nombre de résultats positifs prédits par le classificateur.

$$\boldsymbol{Pr\acute{e}cision} = \frac{Vrai\ positif}{Vrai\ positif + Faux\ positif}$$

- **Rappel (Recall)**

Cette une métrique calcule le nombre de résultats positifs corrects divisé par le nombre de tous les échantillons pertinents.

$$\boldsymbol{Rappel} = \frac{Vrai\ positif}{Vrai\ positif + Faux\ negatif}$$

- **F1 score**

Cette métrique est la moyenne harmonique entre la précision et le rappel. Elle indique la précision du classificateur ainsi que sa robustesse.

$$\boldsymbol{F1} = 2 * \frac{Précision * Rappel}{Précision + Rappel}$$

- **Accuracy**

C'est une métrique qui décrit généralement les performances du modèle dans toutes les classes, c'est le rapport entre le nombre de prédictions correctes et le nombre total de prédictions.

$$\boldsymbol{Accuracy} = \frac{Vrai\ positif + Vrai\ négatif}{Vrai\ positif + Vrai\ négatif + Faux\ positif + Faux\ négatif}$$

5. Conclusion

Au cours de ce chapitre, nous avons présenté la langue arabe, en commençant par présenter ses défis, sa complexité et ses dialectes et le dialecte algérien, sa complexité et sa conjugaison. Nous avons également abordé les principales approches du Deep Learning ainsi que les architecture des modèles que nous allons utiliser pour notre solution finale, le Word embedding et les métriques d'évaluations de ces modèles. Dans le chapitre suivant nous allons parler sur les travaux déjà faits dans le contexte d'analyse de sentiment.

Chapitre III : Travaux Connexes

1. Introduction

Le traitement automatique de la langue arabe n'est pas un sujet récent, il a été étudié et traité dans plusieurs recherches, après avoir vu dans le chapitre deux (02) les particularités du traitement de la langue Arabe ainsi ses dialectes, dans ce chapitre, nous allons exposer des études récentes sur l'Analyse et le traitement de cette langue et qui sont liées à notre projet.

2. Aperçu sur les travaux connexes

Nous allons nous focaliser sur 4 études récentes :

3. Travail 01

Représentation d'encodeur préformée pour le dialecte arabe soudanais.

3.1. Problématique

Quels sont les modèles les plus performants afin d'obtenir de meilleurs résultats sur l'analyse des sentiments soudanais ?

3.2. L'architecture du 1er travail

Dans ce qui suit, nous allons présenter les différents constituants de cette architecture. La figure ci-dessous (figure 25) représente l'architecture général du premier

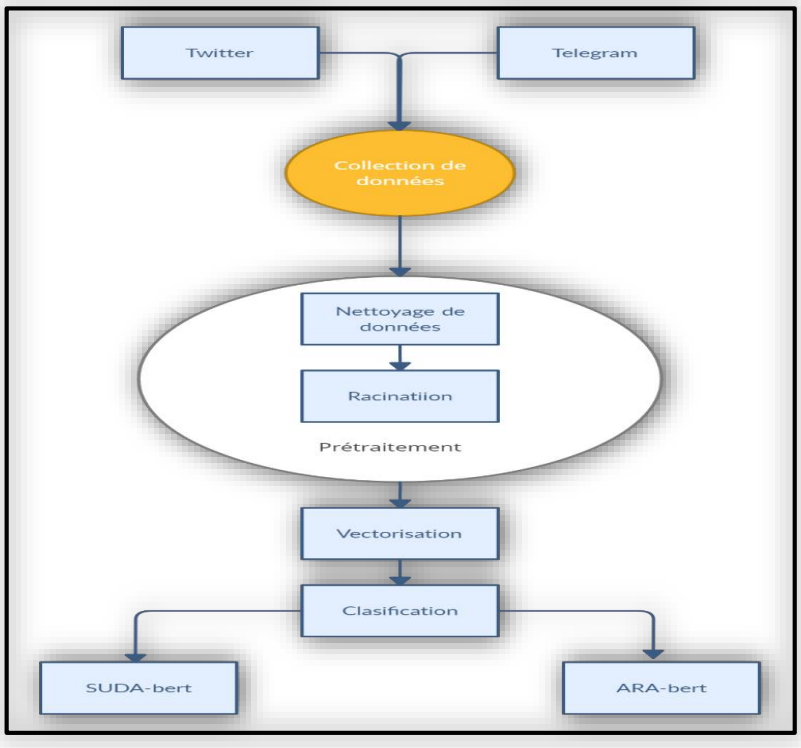


Figure 25: Architecture général du 1er travail

3.3. Ensemble de données

Dans ce travail les auteurs ont collecté à partir du réseau social Twitter et des chaînes publiques Telegram, plus de 7 millions de phrases en dialecte soudanais ou chaque phrase se compose d'au moins 20 caractères.

3.4. Solution proposée

Les différentes tâches de la solution sont brièvement décrites ci-dessous :

- **Prétraitement :** Dans cette phase de prétraitement, une étape de nettoyage des données s'est effectuée en supprimant les symboles vides, Ensuite afin d'effectuer la préformation, ils ont utilisé FARASA qui a servi à la tâche du stemming.
- **Classification :** Dans ce travail afin d'effectuer la classification, les auteurs ont utilisé Arabe-BERT un modèle pré-entrainé sur la langue Arabe ainsi que Suda-Bert un modèle pré-entrainé sur le dialecte soudanais.
- **Résultats :** Les résultats obtenus sont résumés dans le Tableau 10.

Métrique	Arabe-BERT	Suda-BERT
Accuracy	75.4%	76.2%
F1	**57.7%**	**60.6%**

Tableau 14: Résultat d'application d'Arabe-Bert et Suda-Bert sur l'analyse des sentiments

D'après les résultats obtenus nous remarquons que les performances du modèle Suda-Bert sont meilleures que ceux d'Arabe-Bert et cela est dû au pré-entrainement du modèle Suda-Bert sur le dialecte soudanais.

4. Travail 02

Deep Learning pour l'analyse des sentiments du dialecte tunisien.

4.1. Problématique

Quel est la meilleur approches et modèle afin d'analyser les commentaires des internautes postés sur les réseaux sociaux en dialecte tunisien ?

4.2. L'architecture du 2éme travail

la figure ci-dessous (figure 26) représente l'architecture générale de ce travail.

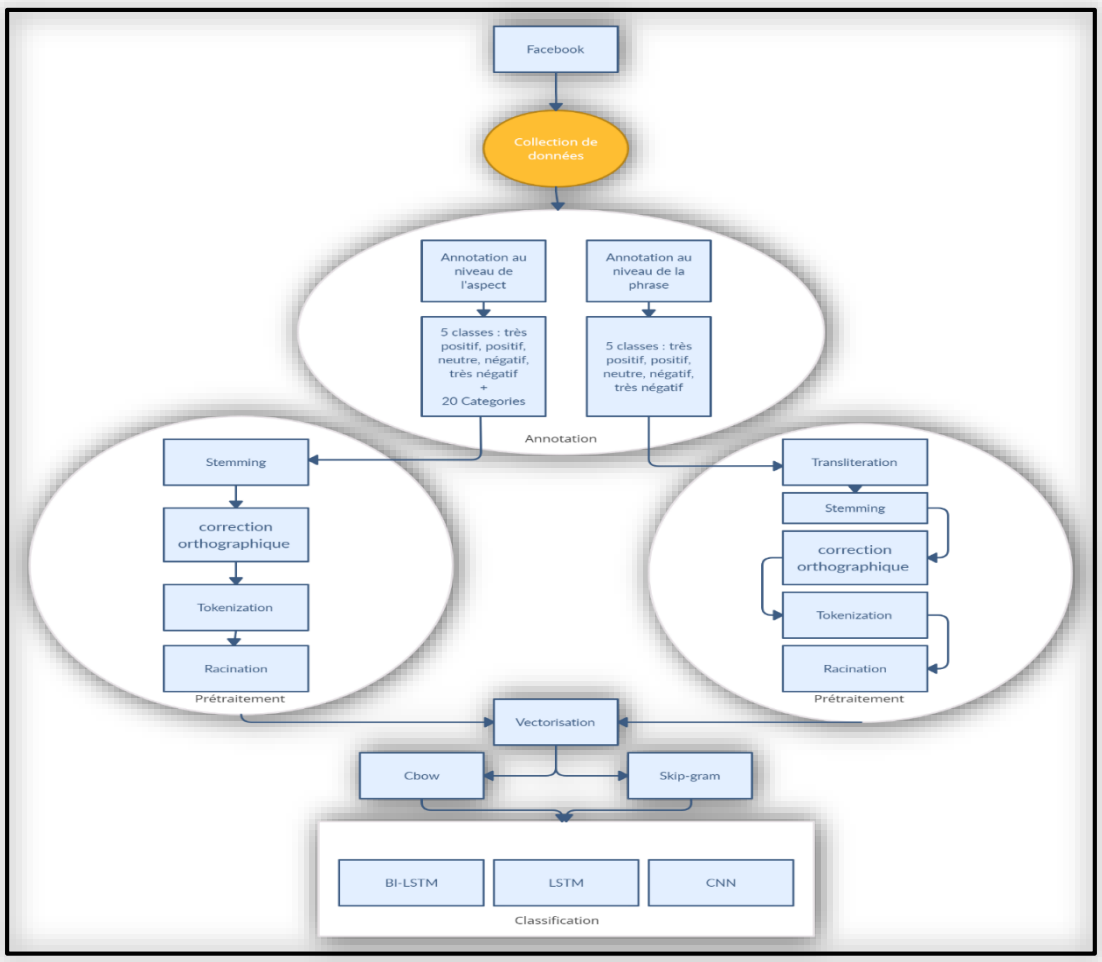


Figure 26: Architecture général du 2éme travail

4.3. Ensembles de données

Dans ce travail les auteurs ont collecté les données à partir des pages Facebook officielles des supermarchés tunisiens des commentaires de deux types : le premier écrit sur la base d'un script latin et le deuxième type écrit en arabe.

4.4. Solution proposée

Les différentes tâches de la solution sont brièvement décrites ci-dessous :

- **Prétraitement :** Pour cette phase, les auteurs ont effectué une séparation des corpus en caractères arabe et latins et ont effectué deux prétraitements différents selon le niveau d'analyse.
- **Vectorisation :** Dans cette phase ils ont implémenté les algorithmes Word2Vec Incluent deux modèles : Skip-gram et le modèles CBOW.
- **Classification :** ils ont sélectionné trois algorithmes : CNN, LSRM, BI-LSTM.

- **Résultats :** Les résultats obtenus sont résumés dans les tables suivantes :

	l'analyse au niveau de l'aspect			l'analyse au niveau de la phrase		
	CNN	**LSTM**	**BI-LSTM**	**CNN**	**LSTM**	**BI-LSTM**
sans supprimer les mots vides et sans stemming	46%	46%	48%	51%	47%	50%
avec suppression des mots vide et effectuant le stemming	47%	46%	49%	51%	47%	50%

Tableau 15: Résultats F-mesure pour l'analyse aux niveaux de la phrase et aspect

	l'analyse au niveau de l'aspect			l'analyse au niveau de la phrase		
	CNN	**LSTM**	**BI-LSTM**	**CNN**	**LSTM**	**BI-LSTM**
Avec 20 catégories et 5 sentiments	47%	46%	49%	51%	47%	50%
Avec 20 catégories et 4 sentiments	42%	33%	40%	65%	58%	60%
Avec 20 catégories et 3 sentiments	48%	47%	47%	58%	47%	55%
Avec 20 catégories et 2 sentiments	42%	41%	41%	77%	75%	78%

Tableau 16:Résultats F-mesure pour l'analyse aux niveaux phrase et aspect avec différentes classes de sentiment et 20 aspects

	l'analyse au niveau de l'aspect			l'analyse au niveau de la phrase		
	CNN	**LSTM**	**BI-LSTM**	**CNN**	**LSTM**	**BI-LSTM**
Avec 8 catégories et 5 sentiments	61%	61%	62%	48%	46%	50%
Avec 8 catégories et 4 sentiments	54%	49%	55%	60%	58%	60%
Avec 8 catégories et 3 sentiments	62%	61%	62%	54%	47%	49%
Avec 8 catégories et 2 sentiments	56%	49%	55%	78%	75%	77%

Tableau 17: Résultats F-mesure pour l'analyse aux niveaux phrase et aspect avec différentes classes de sentiment et 8 aspects

	CNN	LSTM	BI-LSTM
Avec 5 classes	66%	68%	69%
Avec 4 classes	71%	72%	73%
Avec 3 classes	69%	87%	72%
Avec 2 classes	86%	87%	87%

Tableau 18 : Résultats F-mesure pour l'analyse au niveau de la phrase avec différentes classes.

Selon ces résultats, la classification basée sur LSTM et Bi-LSTM ont obtenus les meilleurs résultats pour l'analyse au niveau de la phrase.

5. Travail 03

Analyse des sentiments du dialecte algérien à l'aide d'une approche d'apprentissage en profondeur

5.1. Problématique

Le principal souci est de satisfaire au mieux ses clients abonnés au réseau social Facebook, ou cette analyse se fait par apprentissage profond du dialecte algérien DAlg, où ils ont comparé le modèle CNN avec le classificateur SVM.

5.2. L'architecture du 3éme travail

la figure ci-dessous (figure 27) représente l'architecture générale de ce travail.

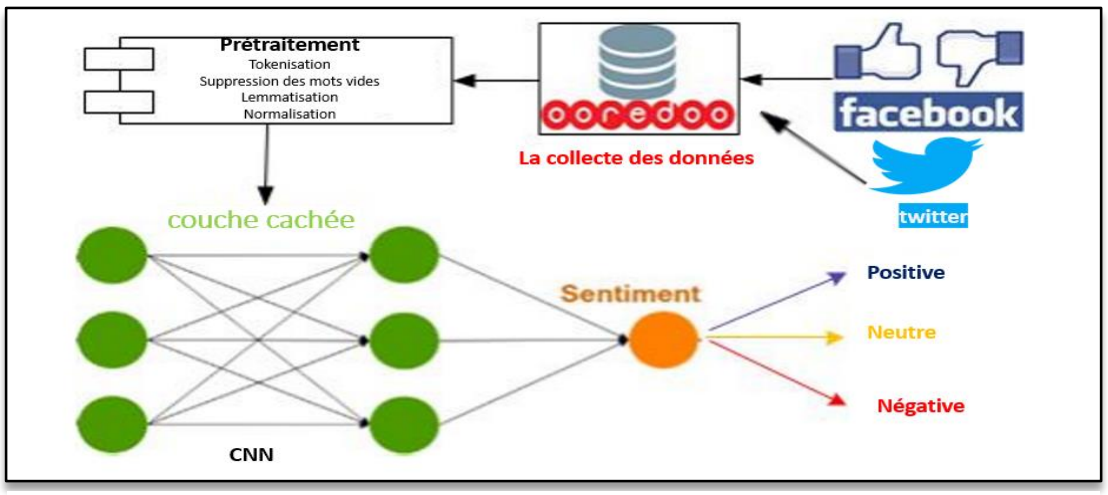


Figure 27: Les étapes Principales de l'approche proposée

5.3. Ensemble des données

Dans ce travail les auteurs ont collecté les données à partir des pages Facebook et Twitter. Leurs corpus contiennent 65125 commentaires.

5.4. Solution proposée

Les différentes tâches de la solution sont brièvement décrites ci-dessous :

- **Détection de langage :** ils ont utilisé la bibliothèque Python Alphabet Detector 11 pour détecter les caractères latins et arabes, afin de créer un corpus spécifique à l'arabe, où ils ont traduit le reste de l'ensemble de données.
- **Nettoyage et prétraitement :** Ils ont commencé par faire la Tokenisation et après ils ont Supprimé les mots vides. Ensuite, ils ont fait la lemmatisation, puis ils ont appliqué la Normalisation lexicale.
- **La classification :** ils ont sélectionné deux algorithmes : CNN et SVM.
- **Résultats :** Les résultats obtenus sont illustrés dans les tableaux ci-dessous en utilisant les trois mesures : précision, rappel et F-mesure.

Le tableau 19 représente les résultats des valeurs de précision pour les classes : positive, négative et neutre de l'ensemble de données.

Précision			
classificateurs	**Positive**	**Négative**	**Neutre**
CNN	0.76	0.72	0.70
SVM	0.72	0.71	0.64

Tableau 19: la Précision des classes positives, négatives et neutres

La table ci-dessous décrit les valeurs de rappel pour chacune des trois classes du classificateur SVM et du modèle CNN.

rappel			
classificateurs	**Positive**	**Négative**	**Neutre**
CNN	0.37	0.81	0.73
SVM	0.24	0.77	0.74

Tableau 20 : le rappel des classes positives, négatives et neutres

La table ci-dessous décrit les valeurs de la F-mesure pour chacune des trois classes pour le classificateur SVM et le modèle CNN.

F-mesure			
classificateurs	**Positive**	**Négative**	**Neutre**
CNN	0.40	0.73	0.68
SVM	0.29	0.72	0.69

Tableau 21: la F-mesure des classes positives, négatives et neutres

La table ci-dessous illustre les résultats expérimentaux obtenus à partir du classificateur SVM et du modèle CNN.

résultats expérimentaux			
classificateurs	**Positive**	**Négative**	**Neutre**
CNN	74.66%	71.00%	67.00%
SVM	69.00%	68.33%	67.66%

Tableau 22: Les résultats expérimentaux obtenu

Les résultats obtenus des tableaux précédents montrent que l'apprentissage en profondeur gère mieux une grande quantité de données par rapport aux algorithmes d'apprentissage automatique, tels que le classificateur SVM.

6. Travail 04

Application de l'analyse des sentiments sur Twitter pour la réputation électronique.

6.1. Problématique

Quels sont les techniques et les algorithmes d'apprentissage automatique utilisées pour détecter la réputation de l'entreprise auprès des internautes, en fonction des trois langues : français, anglais et arabe ?

6.2. L'architecture du 4éme travail

La figure ci-dessous (figure 28) représente l'architecture générale de ce travail.

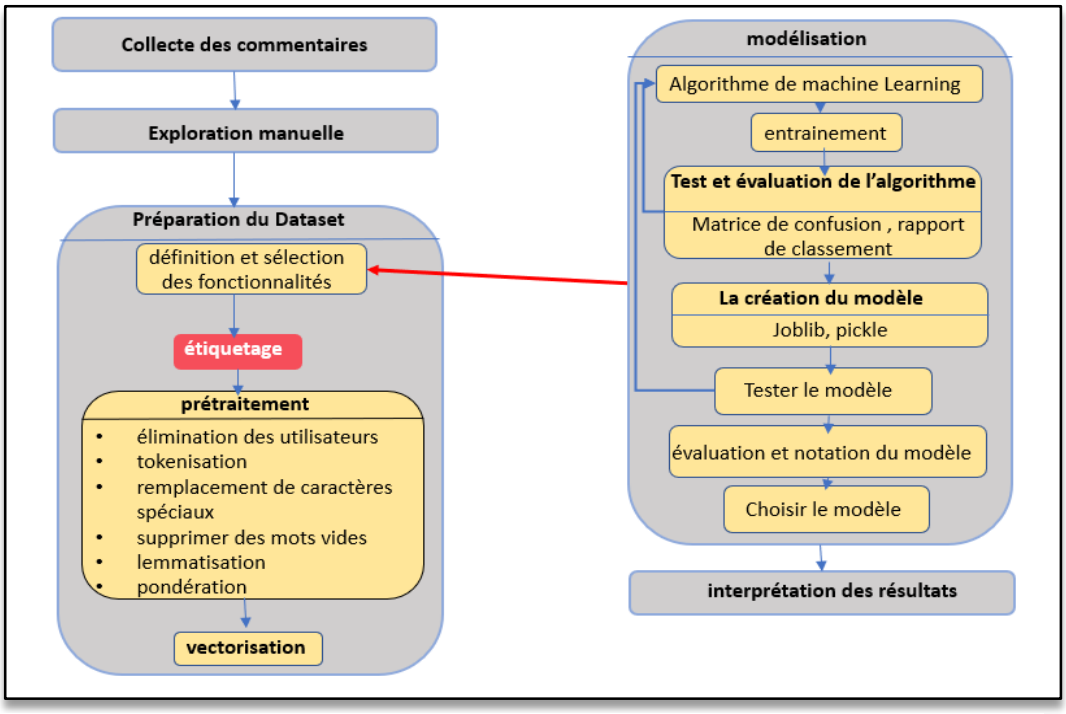


Figure 28: La proposition d'approche

6.3. Ensemble des données

Les auteurs ont utilisés le web scraping[7] pour extraire les commentaires sur Twitter. Dans cette étude ils ont utilisé un ensemble de données de 1510 tweets. Selon l'exploration manuelle, dont 840 tweets positifs et 670 tweets négatifs.

6.4. Solution proposée

Les différentes tâches de la solution sont brièvement décrites ci-dessous :

- **Balayage manuel :** permet de connaître la nature et la langue dans laquelle ils sont écrits.
- **Prétraitement :** Cette étape consiste en quatre tâches :
  - Sélection des fonctionnalités.
  - Étiquetage.
  - Prétraitement des données : Ils ont commencé par faire la Tokenisation et après ils ont Supprimé les mots vides.
- **Vectorisation :** ils ont utilisé l'algorithme du sac de mots. (Bag of Words en Anglais) qui est basé sur deux techniques Count vectorizer et TFIDF Vectorizer.
- **La classification :** Dans cette étape, ils ont décidé d'utiliser 2 algorithmes de Machine Learning, le premier est la régression logistique et le second est SVM (Support Vector Machine).

La table ci-dessous (Table 23) présente la matrice de confusion des pourcentages de tweets trouvés par les algorithmes de Machine Learning (Régression Logistique et SVM) par rapport à l'exploration manuelle.

		Algorithme RL		**Algorithme SVM**	
		positive	**négative**	**positive**	**négative**
Exploration manuelle	**positive**	840	0	830	10
	négative	150	520	120	550

Tableau 23: La Matrice de Confusion

D'après les résultats affichés, la Régression logistique et SVM donnent des valeurs très proches, avec une petite augmentation pour le SVM.

[7] Le web scraping (parfois appelé harvesting) est une technique d'extraction du contenu de sites Web, via un script ou un programme, dans le but de le transformer pour permettre son utilisation dans un autre contexte, par exemple le référencement

7. Analyse comparative des modèles proposées

La table ci-dessous (Table 24) a pour but de comparer les modèles précèdent afin d'extraire le modèle le plus performant et de conclure les résultats.

Etudes	Méthodes	algorithmes	données	résultats
Présentation d'encodeur préformée pour le dialecte arabe soudanais	Apprentissage profond supervisé	SUDA-BERT + ARA-BERT	Commentaires collectés à partir de Twitter et des chaînes publiques Telegram	Résultats du modèle SUDA-BERT meilleures que ARA-BERT avec Accuracy 76,5 et F1 60,6 contrairement à 75,4 et 57,7 pour le modèle ARA-BERT
Deep Learning pour l'analyse des sentiments du dialecte tunisien	Apprentissage profond supervisé	CNN + LSTM + BI-LSTM	Commentaires collectés à partir des pages Facebook officielles des supermarchés tunisiens	Quatre tests ont été effectués avec différents prétraitements, classes et analyse, Le modèle CNN a obtenu les meilleurs résultats pour l'analyse au niveau de la phrase et BI-LSTM pour le niveau d'aspect
Analyse des sentiments du DAlg a l'aide d'une approche d'apprentissage en profondeur	Apprentissage profond supervisé + apprentissage automatique supervisé	CNN + SVM	Commentaires extraits du Facebook et twitter collectées et annotées à l'aide de l'api Facepager et Tweepy	le modèle CNN atteint une précision 74.66% qui est plus élevée par rapport à celle obtenue par le SVM 69.00%
Application d'analyse des sentiments pour La réputation électronique	apprentissage automatique supervisé	Régression logistique + SVM	commentaires des différents réseaux sociaux en utilisant le web scraping pour twitter	La Régression logistique et SVM donnent des valeurs très proches avec une petite augmentation pour le SVM.

Tableau 24: tableau comparative des quatre travaux connexes

8. Conclusion

Dans ce chapitre, nous avons présenté quelques travaux connexes à notre problématique et nous avons décrit les techniques qu'ils utilisées dans l'analyse des sentiments de différents dialectes arabe. Dans le chapitre qui suit, nous allons voir la conception de notre processus pour la multi classification automatique du dialecte algérien.

Chapitre IV : Conception et modélisation de la solution

1. Introduction

Dans ce chapitre, nous allons d'abord présenter le corpus sur lequel nous avons travaillé, notre vision conceptuelle pour le système d'Analyse de Sentiments pour le dialecte Algérien en expérimentant les approches Deep Learning.

Nous avons suivi la méthodologie illustrée ci-dessous, présentant la conception de notre processus en commençant par la collecte des données, le prétraitement ainsi la classification à l'aide des modèles Deep Learning.

2. Rappel de la problématique

L'application de l'Analyse des Sentiments sur les opinions des clients de l'entreprise Djezzy rédigés en dialecte Algérien est un intérêt récent dans l'entreprise. De plus, vu le volume grandissant de données de jours en jours et de mois en mois, la nécessité de mettre en place une solution intelligente pour le traitement, l'Analyse et la classification efficace de ces commentaires s'est fait sentir car parallèlement à leurs tâches quotidiennes, les opérateurs, personnels, et administrateurs de l'entreprise n'ont pas beaucoup de temps et d'efforts pour s'atteler à cette tâche faramineuse et délicate. Dans les sections qui suivent, nous allons décrire notre solution à ce problème et ce, depuis les stades primaires de constitution du Dataset, des tests à travers différents modèles de Deep Learning, jusqu'à l'élaboration de l'outil final d'Analyse et de suivi des avis des clients sur diverses offres et services.

3. Processus global de la solution

Ce processus débute par l'étape de collecte de données, puis vient l'étape de l'annotation des commentaires pour pouvoir faire les tests ultérieurs. Ensuite, vient l'une des étapes cruciales qui est la formation des modèles. Par la suite, vient la classification des données et leur visualisation graphique.

Nous présentons dans la figure ci-dessous (figure 29) le processus que nous avons a suivi pour l'élaboration de notre solution, les détails de chaque étape seront présentés ci-après.

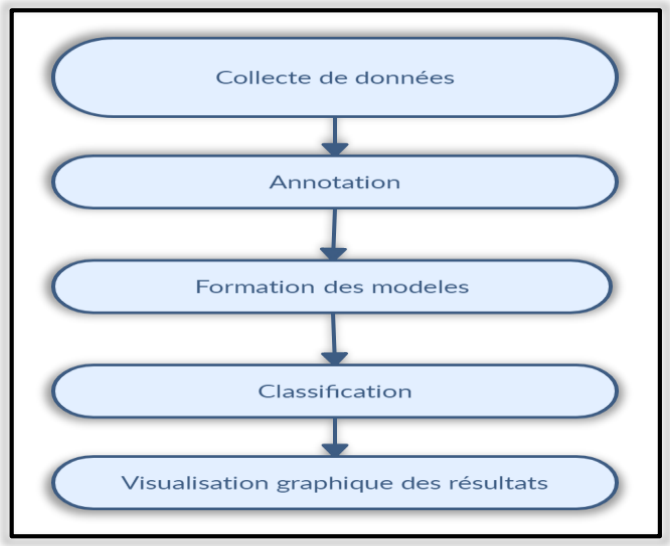


Figure 29: Processus du travail

4. La collecte des données

index	comment
0	لاه السرعة في جيزي فرق بين هايلة بزاف وهايلة ماكسي
1	4 وهل هناك جديد في مايخص المودم g lte
2	13 وش قلكست 100تلف عطوني 30 عاود قلولي بلي عطوني
3	ردونا في تدفق المنخفض
4	ya djezzy wlh ila bzzaf 3liha tkhfaf 30 min tat9al 5h
5	شكرا ا
6	pixx لازم تهجمو على صفحة موبيليس في تعليقات تطالب بإلغاء العرض او امكانية الرجوع للعرض القديم
7	djezzy confort fiha limitation
8	اللهم من ولي من أمر أمتي شيئا فشق عليهم فاشقق عليه ومن ولي من أمر أمتي شيئا فرفق بهم فارفق به
9	vive la switch
10	مازلت لا أفهم لماذا الشركات عددنا متخلفة الى هذه الدرجة و نحن في 2020 و الى متى هكذاالكبر دولة في افريقيا و ثرواتها كبيرة و مازلنا في هذا المستوى المتدني حقا لم اعد اتحمل هذه الاحوال في هذا البلد اللعين
11	موبيليس خير من جيزي الانترنت طيارة بوضع طروا جي و الكانجي
12	السرعة تاعو ضعيفة بزاف

Figure 30 : Aperçu sur la collection de données

La première étape de notre processus d'Analyse des Sentiments a consisté à collecter des commentaires en dialecte Algérien, cela a été fait à partir de différentes sources.

- **La première source utilisée :** le site web d'hébergement de vidéos et média social YouTube. Nous avons effectué cette tâche en extrayant les commentaires des vidéos postées par des présentateurs d'offres des opérateurs téléphoniques Algériens. Notre stratégie a consisté a utilisé l'API YouTube en créant des comptes développeurs de google, installant les bibliothèques requises sur python afin de pouvoir extraire les commentaires de chaque vidéo avec son identifiant.

Par la suite une phase de filtrage des commentaires a été effectuée en gardant uniquement les commentaires objectifs et pertinents.

- **La deuxième source utilisée :** le réseau social Facebook, en effectuant une collecte manuelle des commentaires à partir des pages Facebook des trois opérateurs téléphoniques Djezzy, Mobilis, Ooredoo.

Après la collecte, nous avons obtenus 6643 commentaires réparties en quatre types : le dialecte Algérien écrit en caractères Arabe, l'arabizi écrit en caractères latins, le Français et un mixte entre le français et l'arabizi, La Figure ci-dessous représente un aperçu de notre collection de données.

4.1. Annotation

L'annotation manuelle des commentaires est une tâche couteuse en termes de temps. Cette dernière nous a pris jusqu'à trois semaines (03) et en s'y mettant) deux pour la collecte et l'annotation. A l'issu de cette étape, nous avons annoté les 6643 commentaires en trois polarités : **positive**, **négative** et **neutre**.

Le tableau ci-dessous représente un exemple de 5 commentaires de notre collection de données annotées en trois polarités. Telles que le 0 correspondrait à un commentaire neutre, le 1 correspondrait à un commentaire négatif et enfin, le 2 correspondrait à un commentaire positif.

Commentaire	Annotation
الانترنت بتدفق مليح	**2**
راني غير نفليكسي تدوهالي علاه منكر	**1**
هل هي متوفرة حاليا	**0**

Tableau 25: Exemple d'annotation de commentaires

La figure ci-dessous représente la distribution de nos commentaires qui est de 2347 neutres, 3003 négatifs, 1393 positifs

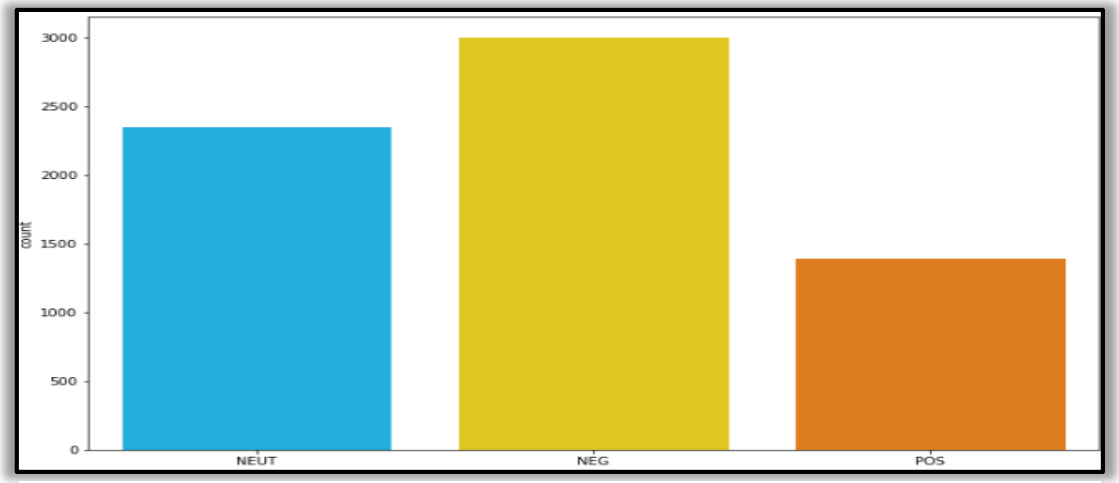


Figure 31: Distribution des commentaires

5. Architecture de prétraitement

La figure suivante représente la solution qu'avons proposée pour le traitement du dialecte algérien.

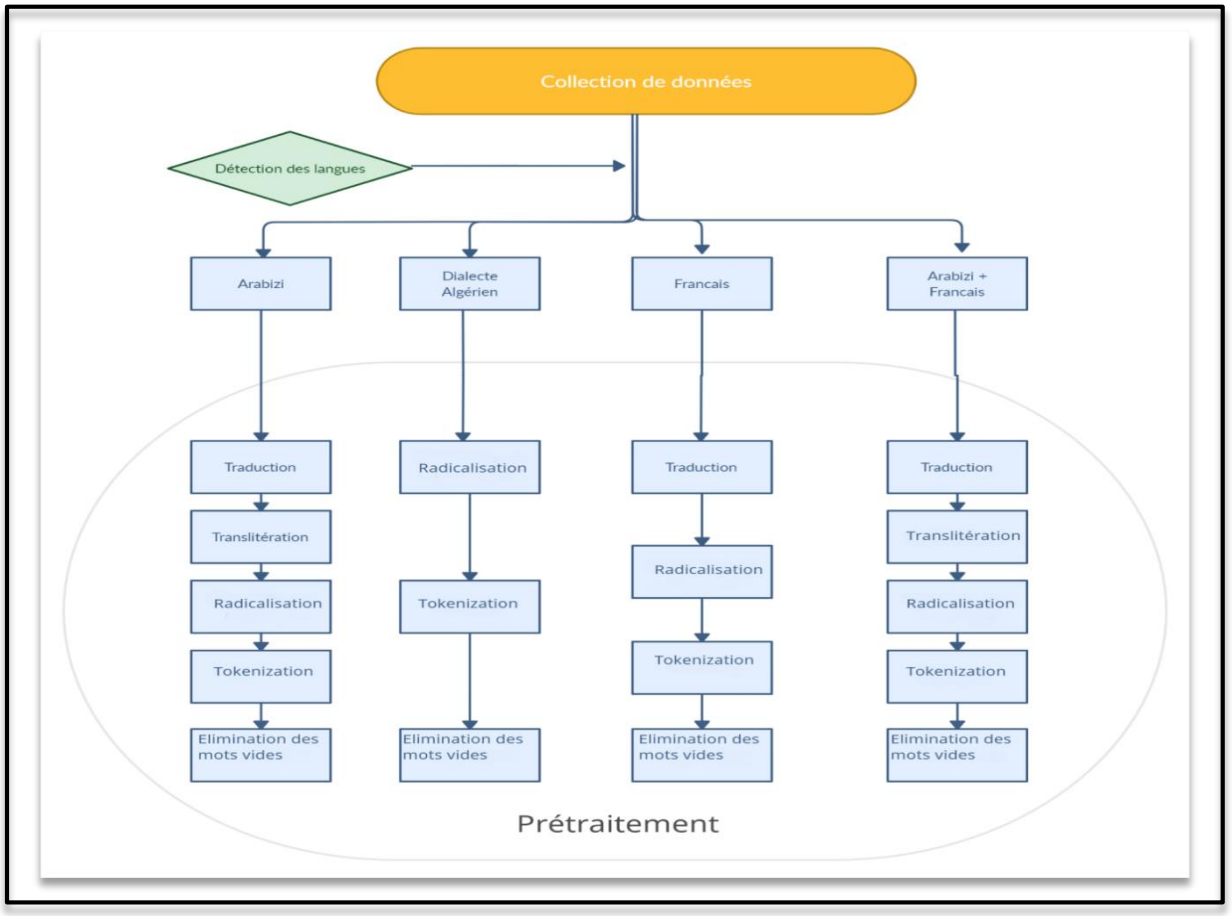


Figure 32: Architecture pour le traitement du dialecte algérien

Ci-dessous, nous allons définir chacune des tâches illustrées dans la figure précédente.

5.1. Détection de la langue

Comme le montre la Figure précédente, nous commençons notre traitement par la détection de la langue, nous avons élaboré cette solution vue l'existence de quatre types de commentaires dans notre corpus. De ce fait, notre approche diffère par rapport aux autres travaux déjà réalisés que nous avons cité dans le chapitre précèdent. En plus, d'être plus riche car acceptant des commentaires Algériens dans 4 types d'écritures différents, elle promet d'être moins coûteuse en termes de temps.

Texte	Langue détectée
C'est une bonne offre 3ajbetni mais le débit est faible	Français
راني نخدم بيها ننصحكوم متشروهاش ثقيلة بزاف	Arabe

Tableau 26 : Exemple de détection de langue sur un texte

5.2. Traduction

Dans cette phase, précisément pour les commentaires rédigés en Arabizi, en Français ou un mixte entre l'Arabizi et le Français, nous effectuons la traduction des termes détectés en Français vers leur équivalent Arabe en utilisant une approche linguistique.

Figure 33: Exemple de traduction sur un texte

5.3. Translitération

Cette phase est reliée à la précédente, pour chaque terme non traduit nous lui effectuons une translitération. Elle consiste à remplacer chaque caractère latin par son caractère arabe.

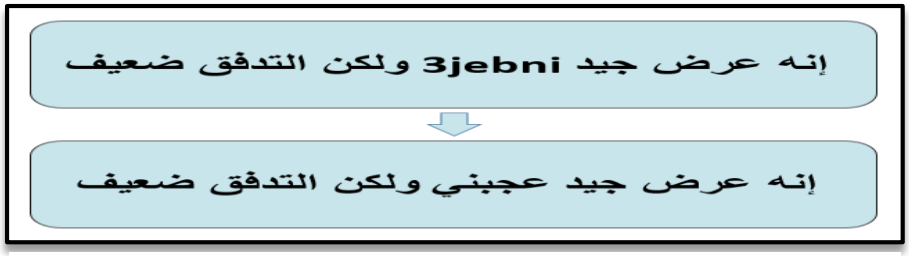


Figure 34: Exemple de translitération sur un texte

5.4. Radicalisation

La radicalisation est un processus de réduction des mots écrit dans différentes formes vers leur racine originale.

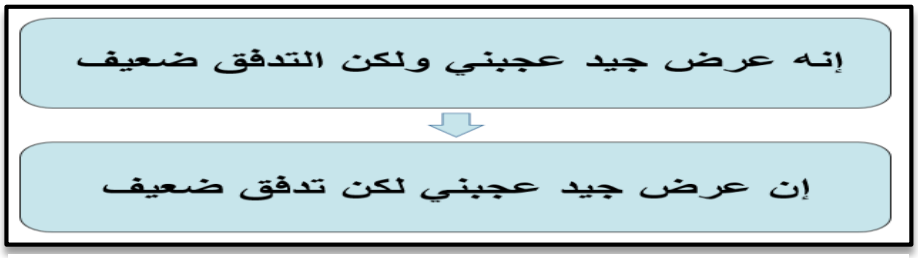


Figure 35: Exemple de radicalisation sur un texte

5.5. Tokenisation

Cette phase vise à deviser le texte en mots (jetons). Cela permet d'interpréter le sens du texte en analysant la séquence des mots.

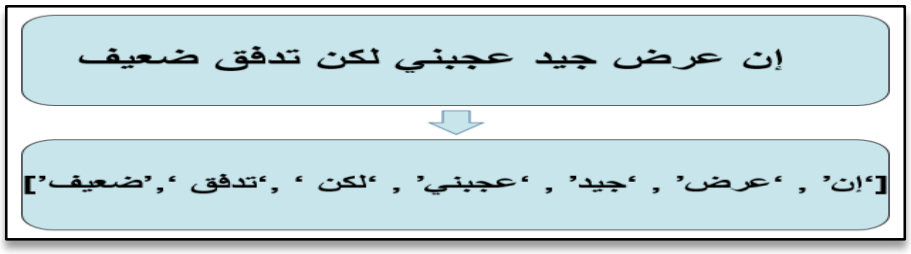


Figure 36: Exemple de tokenisation sur un texte

5.6. Elimination des mots vides

Un mot vide est un mot qui est couramment utilisé mais qui n'est pas important en termes d'information ni ne possède, dans notre cas, un effet quelconque sur la classification. Dans notre travail, nous avons effectué cette élimination à partir d'une liste de mots vides du dialecte algérien qui contient environ 1000 mots.

Exemples : هيا, غير, أنا, ما, بعدا …

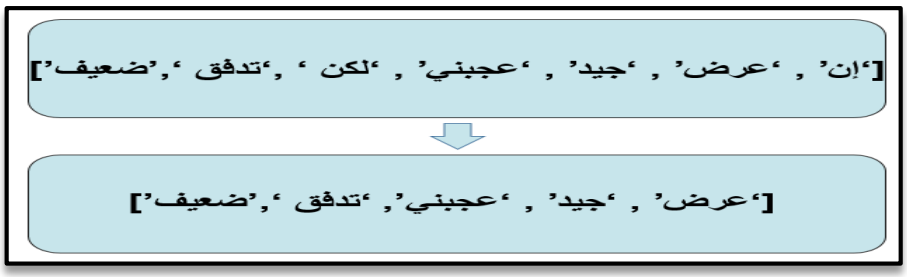


Figure 37: Exemple d'élimination de mots vides sur un texte

Mais il faut toutefois dans cette étape prendre en considération le fait que certains mots considérés comme vides ou non importants dans certaines applications peuvent être indicateurs de subjectivité ou de polarité. Ceci peut fausser les résultats.

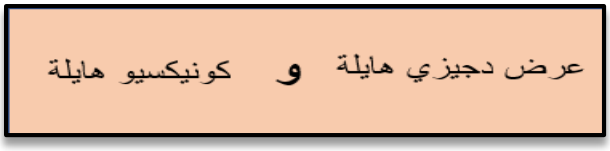


Figure 38: Exemple d'un cas particulier des mots vide

6. Formation des modèles

Une fois, nos données collectées, annotées et prétraitées, nous avons opté pour cinq algorithmes basés sur le Deep Learning ou l'apprentissage en profondeur.

6.1. L'architecture du système

Avant de détailler la démarche de notre travail, nous représentons dans la figure si dessous l'architecture globale de notre solution pour l'analyse des sentiments.

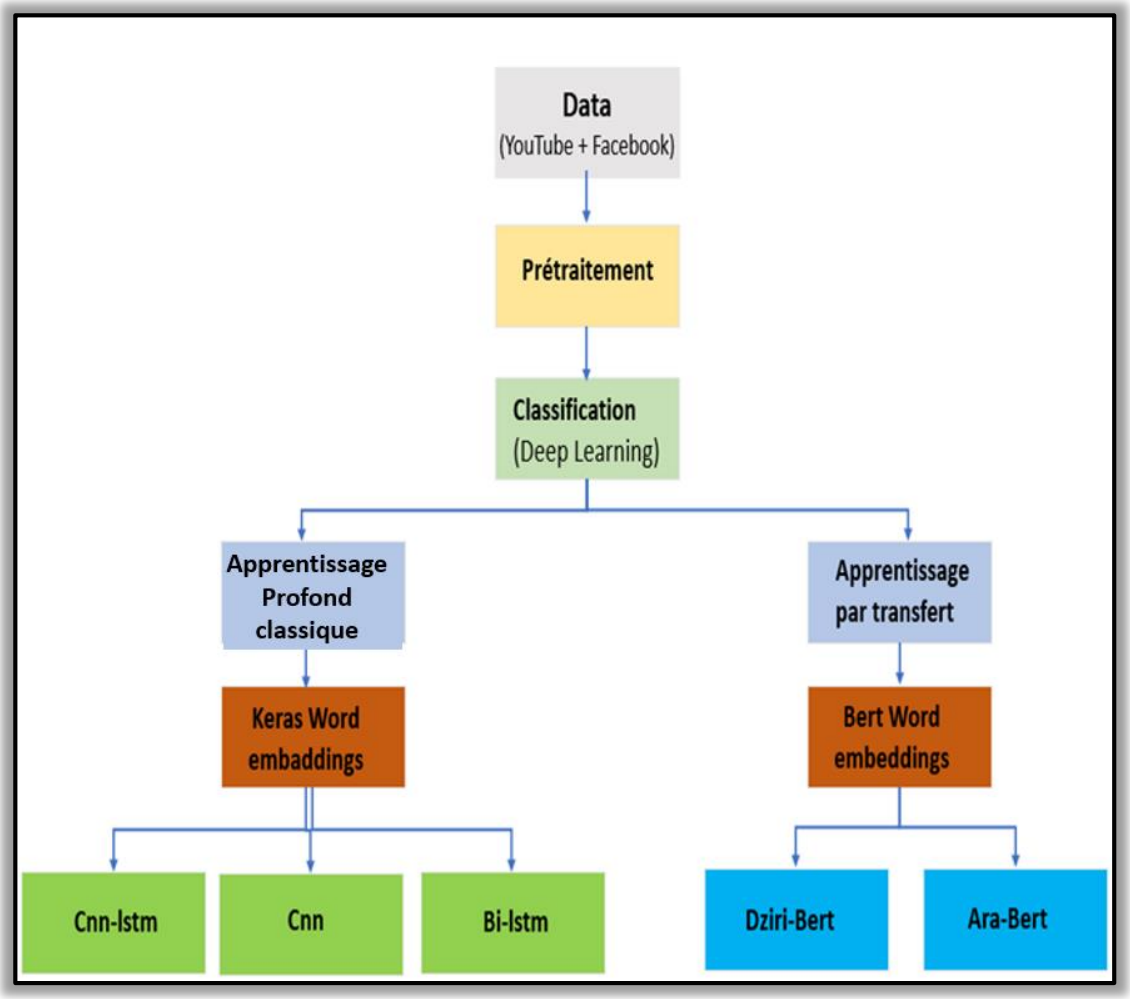


Figure 39: L'architecture globale de notre système d'analyse de sentiment

6.2. Solutions proposées

Nous avons opté pour deux types d'approche de Deep Learning. La première se basant sur l'apprentissage profond classique et la seconde se basant sur l'apprentissage par transfert.

6.2.1. Classification avec apprentissage profond classique

Avant le processus d'apprentissage, les textes de notre corpus doivent être convertis en vecteurs numériques, pour cela nous avons utilisé keras Word embading et pour la classification nous avons utilisé 3 modèles (CNN-Lstm, Bi-lstm, cnn) :

6.2.1.1. Keras Word embedding

Dans cette section, nous avons vectorisé des ensembles de données d'entraînement et de test afin qu'ils puissent être transmis directement au réseau de neurones. Afin de vectoriser les données, nous avons utilisé **Tokenizer ()** disponible sur **keras**.

Nous avons d'abord créé une instance de **Tokenizer**, puis appelé la méthode **Fit_on_texts ()** avec nos ensembles de données (train et test) pour remplir le vocabulaire avec des jetons. La méthode tokenisera chaque commentaire et en ajoutera des jetons au vocabulaire.

Une fois que nous avons rempli le vocabulaire, nous pouvons traduire n'importe quel commentaire en une liste d'index en appelant la méthode **Texts_to_sequences ()** sur l'objet **Tokenizer**. La méthode prend une liste de commentaires en entrée et renvoie une liste d'indexes.

Le processus du Word embedding est comme suit :

- Tout d'abord, nous allons vectoriser les données textuelles avec la méthode Tokenizer ().

- Ensuite, Nous allons appliquer une méthode de remplissage pour ajouter des zéros et définir la taille fixe dans chaque vecteur.

- nous allons charger des données textuelles et les diviser en parties d'apprentissage et de test.

6.2.1.2. Définir le réseau

Dans cette section, nous avons défini les 3 modèles de classification de texte à l'aide de l'API **séquentielle de Keras**. Les 3 modèles contiennent des composants essentiels comme résumé dans le tableau suivant :

Hyper paramètre	Le rôle
Embedding ()	Transforme les entiers positifs (index) en vecteurs denses de taille fixe
input_length	spécifie le nombre de jetons par exemple le nombre de mots du texte.
embed_Dim	C'est la taille des vecteurs que le modèle génère.
Vocab_size	L'ensemble des mots uniques utilisés dans le corpus de texte
Dense	couche dense effectue l'opération sur l'entrée et renvoie la sortie.
Dropout	Il est utilisé pour résoudre le problème de surajustement.

Tableau 27: Hyper paramètres utilisées et leurs rôles

6.2.1.3. Architecture des modèles (CNN-Lstm, Bi-lstm, cnn)

Avant d'expliquer le processus de l'entrainement des modèles, nous présentons son architecture qui est la suivante.

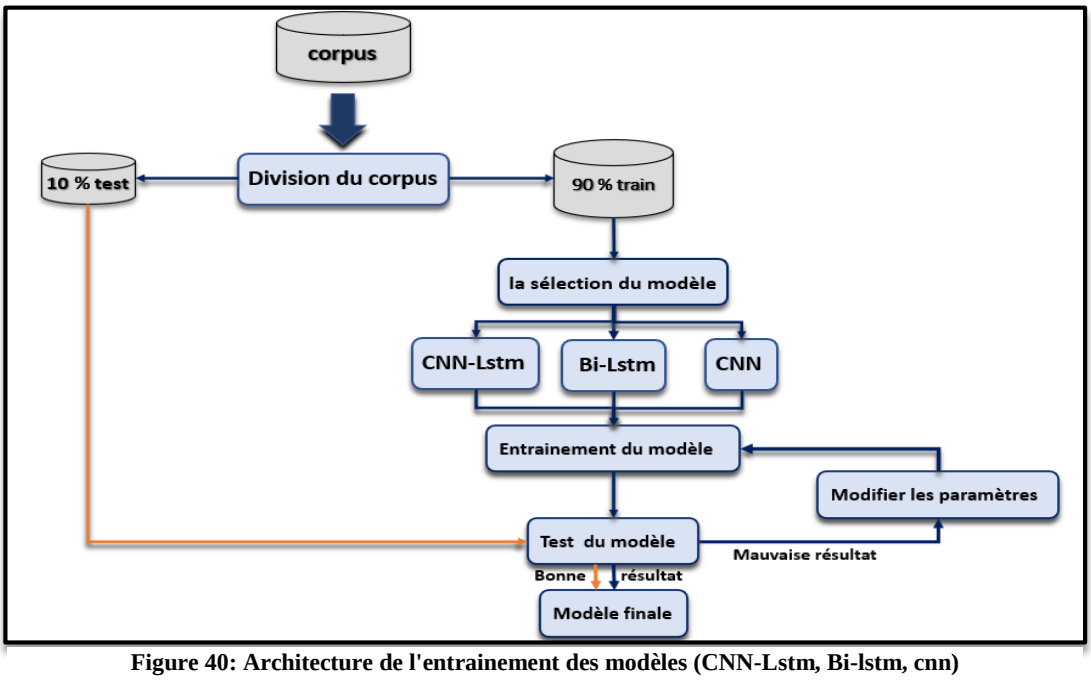


Figure 40: Architecture de l'entrainement des modèles (CNN-Lstm, Bi-lstm, cnn)

La figure ci-dessus montre le processus d'entrainement des modèles cnn-lstm, Bi-lstm, cnn est c'est comme suit :

- Diviser Dataset en train et test.
- En utilisant les données train pour entrainer notre modèle
- Utiliser les Word embading pour convertir les données catégorielles en numérique
- La sélection de notre modèle.
- Si le modèle n'a pas donner des bons résultats on va régler le paramétrage du modèle.
- Sinon on va utiliser les données test pour évaluer notre modèle

6.2.2. Solutions proposées basant sur l'apprentissage par transfert

6.2.2.1. Bert Word Embading

Avant le processus d'apprentissage, les textes de notre corpus doivent être convertis en vecteurs numériques, pour cela nous avons utilisé deux modèles basés sur l'architecture BERT (DZIRI-Bert et ARA-Bert).

6.2.2.2. Définir les modèles

- **DZIRI-BERT :** un modèle de langage basé sur Transformer qui a été préformé spécifiquement pour le dialecte algérien (1 million de tweets). Ce modèle gère le contenu des textes algériens écrits en caractères arabes et latins [29].
- **ARA-BERT :** un modèle de langue basé sur Transformer qui a été préformé pour la langue arabe (70 millions de phrases). Il existe en deux versions : Ara-BERT V1 et Ara-BERT V2, dans notre produit nous avons utilisé la deuxième version qui est préformé avec un meilleur vocabulaire et plus de données.

Avant d'expliquer le processus de l'entrainement des modèles, nous présentons son architecture qui est la suivante.

Le processus d'entrainement les modèles DZIRI_Bert et ARA_bert est comme suit

- Diviser les données en train et test.
- En utilisant les données train pour entrainer notre modèle
- La sélection de notre modèle (DZIRI_Bert ou ARA_bert).
- Si le modèle n'a pas donné des bons résultats on réglera le paramétrage du modèle.
- Sinon on va utiliser les données test pour évaluer notre modèle.

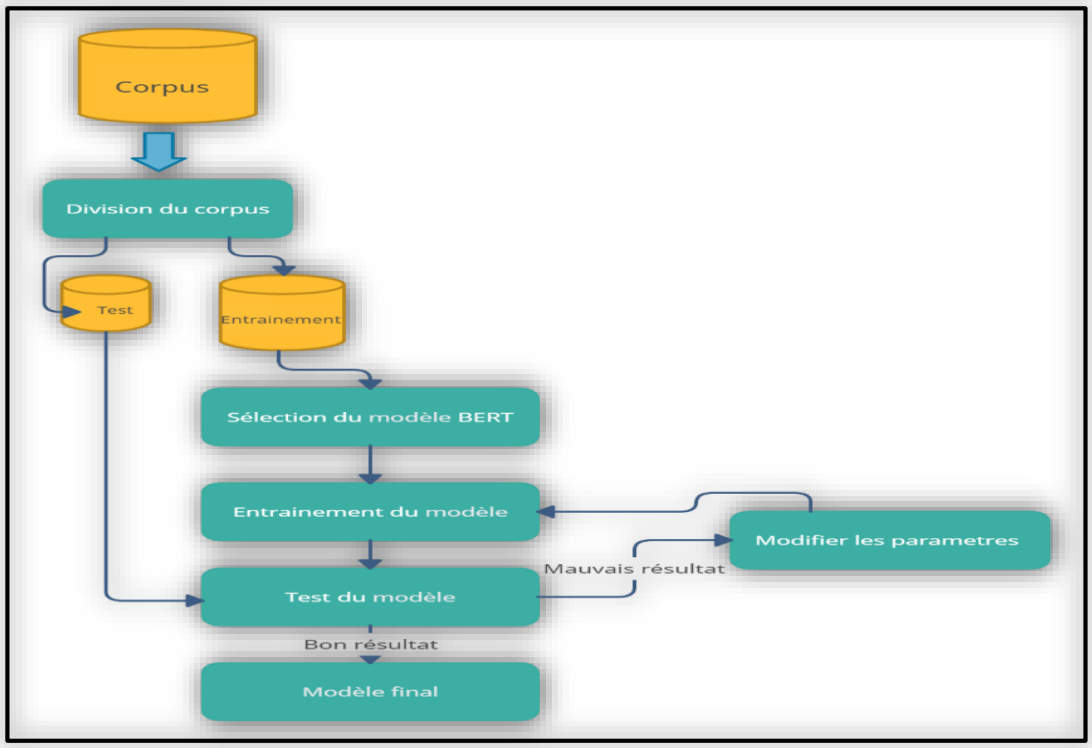


Figure 41: Architecture de l'entrainement des modèles BERT.

Avant d'effectuer l'entrainement sur les modèles nous devons d'abord avoir deux corpus disponibles, le premier sera utilisé pour la phase d'entraînement des modèles, le deuxième sera utilisé pour la phase de test, de sorte que nous allons diviser notre corpus initial une fois prétraité en 80% pour la partie entraînement et 20% pour la partie de test. Ensuite, on effectue la sélection du modèle souhaité (DZIRI-BERT ou ARA-BERT), le réseau de neurones sera formé sur la partie d'entraînement et sera validé à partir de la partie du test. Si le modèle obtient de bons résultats il sera validé sinon on effectuera une modification sur les paramètres du réseau neuronal afin d'obtenir les résultats souhaités.

7. Conclusion

Dans ce chapitre, nous avons vu notre approche globale pour développer un modèle d'Analyse de Sentiments pour le dialecte Algérien en mettant l'accent sur les algorithmes de classification. Dans le prochain chapitre nous allons voir l'implémentation de notre solution, les modèles utilisés et les résultats obtenus ainsi que leur analyse. Finalement, nous allons présenter l'interface finale de notre système d'Analyse de Sentiment du dialecte Algérien des commentaires des clients de Djezzy Télécom.

Chapitre V : Implémentation de la solution

1. Introduction

Dans ce chapitre, nous allons d'abord présenter les outils et librairies que nous avons utilisées pour implémenter notre projet. Ensuite nous allons parler de l'implémentation de notre solution en commençant par le prétraitement jusqu'à la classification. Par la suite, nous analyserons les résultats des tests effectués et enfin, nous présenterons l'interface de visualisation de l'Analyse de Sentiments des commentaires des clients de Djezzy selon le meilleur modèle de classification retenu.

2. Environnement de travail

Pour la réalisation de notre projet, nous avons utilisé un ensemble d'outils et ressources que nous allons décrire ci-après :

2.1. Environnement logiciel

Dans cette section, nous allons parler de notre environnement logiciel. En passant en revue, le langage de programmation, les plateformes, ainsi que les APIs.

2.1.1. Python

Python est un langage de programmation open source. Il dispose de structures de données de haut niveau et permet une approche efficace de la programmation orientée objet. C'est un langage de programmation qui permet plusieurs utilisations aux développeurs tel que : l'Intelligence Artificielle, la programmation d'application, le web... [31]. Ces dernières années, Python est devenu l'un des langages de programmation de prédilection pour la communauté TAL.

Figure 42: Logo Python

2.1.1. Jupiter

Jupiter est un environnement de développement interactif utilisé pour faire de la programmation dans plusieurs langages dont Python. Son interface permet de modifier et exécuter les documents depuis le navigateur.

Figure 43: logo de Jupiter

2.1.1. Google colab

Google Colab est un outil de Google destiné à la formation et à la recherche dans l'apprentissage automatique**.** Il permet la formation des modèles de machine Learning et Deep Learning directement dans le navigateur.

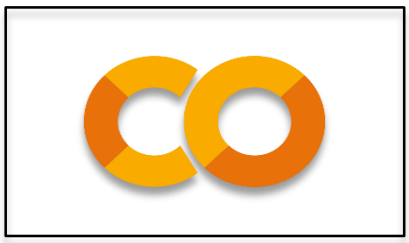


Figure 44: Logo Google colab

Nous allons maintenant parler des librairies utilisées dans ces environnements de travail.

2.1.2. Scikit Learn

Scikit learn est une bibliothèque du langage de programmation python destinée à l'apprentissage automatique, elle propose des outils pour la classification, le prétraitement, la régression …, c'est une bibliothèque très importante et utile dans l'apprentissage automatique.

Figure 45: Logo Scikit learn

Nous avons utilisé cette bibliothèque afin de diviser notre corpus en sous-ensembles d'entraînement et de tests et aussi pour calculer les métriques lors de l'entrainement des modèles.

2.1.1. Matplotlib

Matplotlib est une bibliothèque du langage de programmation python, elle permet diverses utilisations telles que la création de visualisations statiques, animées et interactives.

Elle permet d'intégrer ces visualisations dans des applications et les visualiser sous forme d'interface graphique de python.

Figure 46: logo de Matplotlib

Nous avons utilisé cette bibliothèque afin visualiser la distribution des commentaires de notre corpus d'entrainement sous formes de graphiques, ainsi que les résultats numériques des modèles entrainés.

2.1.2. PyTorch

PyTorch est une bibliothèque du langage de programmation python qui permet d'effectuer des calculs tensoriels pour l'apprentissage profond, son objectif est de permettre l'implémentation et l'entraînement de modèles.

Figure 47: Logo PyTorch

Nous avons utilisé cette bibliothèque afin d'optimiser les calculs tensoriels par le GPU de Google colab.

2.1.3. TkinTer

Tkinter est une bibliothèque du langage de programmation python qui permet la création des interfaces graphiques.

Figure 48: Logo TkinTer

Nous avons utilisé cette bibliothèque afin de réaliser l'interface de notre application finale de visualisation des avis des clients Djezzy. **Power BI**

Power Bi est un logiciel de Microsoft qui permet d'effectuer une visualisation de données avec une interface simple et compréhensible pour les utilisateurs finaux.

Figure 49: Logo Power BI

Nous avons utilisé ce logiciel afin de réaliser une sorte de tableaux de bords synthétisants les résultats obtenus une fois la classification effectuée.

2.2. Environnement matériel

Nous avons utilisé dans notre travail deux ordinateurs dont les caractéristiques sont résumées dans le tableau suivant :

	Machine 1	**Machine 2**
Modèle	Asus Vivobook	Lenovo thinkbook
Processeur	Intel Core i5 8th Gen	Intel Core i5 11th Gen
Type du système	Windows 10 Système d'exploitation 64bits	Windows 11 Système d'exploitation 64bits
RAM	4GO	8GO

3. Mise en œuvre

Dans cette partie nous allons reparler de la solution que nous avons proposé dans le chapitre précèdent. En nous intéressant cette fois, aux détails techniques.

3.1. Prétraitement

Nous allons présenter l'implémentation du prétraitement que nous avons effectué.

3.1.1. Détection de la langue

Nous avons utilisé l'API de Google traduction. Les commentaires détectés en « Fr » sont des commentaires en français, « Ar » pour les commentaires en arabe et « autres » pour l'arabizi et le français mixte avec l'arabizi.

La figure ci-dessous présente le code utilisé pour la détection de la langue sur notre corpus.

```
from langdetect import detect

def det(x):
    try:
        lang = detect(x)
    except:
        lang = 'Other'
    return lang
```

Figure 50: Code source utilisé pour la détection de la langue

Une synthèse des résultats obtenus après exécution de ce code est présentée dans la figure ci-dessous.

	comment	note	langue
0	... بالنسبة لجماعة هايلة يزاف عرض 1200الف اكتيفيه	0.0	ar
1	نحوها	3.0	ar
2	gماكسي بيسك يزاف مافيهاش 20	1.0	ar
3	...اذا عندي هايلة و الماكسي و في 2 في حالةالحل ال	3.0	ar
4	هههههه جيت هنا	3.0	fa
...	...	...	...
8440	C est bien mais concernant le débit sincèreme...	1.0	fr
8451	...راني نخدم ليها نتصحكوم متشروهاش تقيلة يزاف على	1.0	ar
8477	صبيب راذا ف 2022 بياتو جابوهم من 2013	1.0	ar
8483	...كونكسيو في حالة وهوما بيلعطو ريقلو الريزو نتعك	1.0	ar
10046	تقييمي 310	1.0	ar

Figure 51: Implémentation de la détection de la langue

3.1.2. La Traduction

Nous avons utilisé aussi l'API de Google traduction pour cette tâche. Nous avons convenu de traduire les termes français vers l'arabe standard.

La figure ci-dessous présente le code utilisé pour la détection de la langue sur notre corpus.

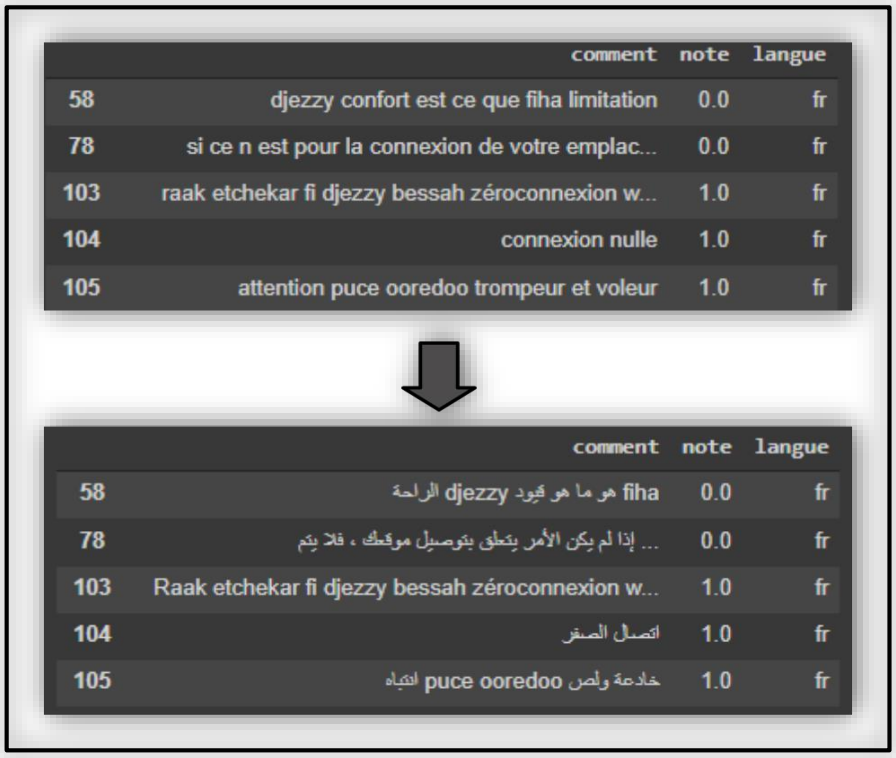
```
from googletrans import Translator
translator = Translator()

def tr(tx):
    rs=translator.translate(tx, src='fr',dest='ar')
    return rs.text
```

Figure 52: Code source utilisé pour la traduction des mots

Une synthèse des résultats obtenus après exécution de ce code est présentée dans la figure ci-dessous

3.1.3. La translitération

Nous avons implémenté la translitération à partir d'une liste qu'on a créé, qui comporte la translitération de chaque caractère latin vers son caractère arabe, aussi une translitération pour les termes écrits en langage SMS et les termes avec des fautes d'orthographe.



	comment	note	langue
58	djezzy confort est ce que fiha limitation	0.0	fr
78	si ce n est pour la connexion de votre emplac...	0.0	fr
103	raak etchekar fi djezzy bessah zéroconnexion w...	1.0	fr
104	connexion nulle	1.0	fr
105	attention puce ooredoo trompeur et voleur	1.0	fr

	comment	note	langue
58	هو ما هو قيود djezzy الراحة fiha	0.0	fr
78	... إذا لم يكن الأمر يتعلق بتوصيل موقعك ، فلا يتم	0.0	fr
103	Raak etchekar fi djezzy bessah zéroconnexion w...	1.0	fr
104	اتصال الصفر	1.0	fr
105	خادعة ولص puce ooredoo انتباه	1.0	fr

Figure 53: Implémentation de la traduction

La figure ci-dessous présente un extrait de notre liste de translitération

```
"a": u"ا","b": u"ب", "c": u"ك", "d": u"د", "e": u"ا", "f": u"ف", "g": u"ق", "h": u"ه","i": u"ي", "j": u"ج", "k": u"ك", "l": u"ل",
"m": u"م","n": u"ن", "o": u"و", "p": u"ب", "q": u"ق","r": u"ر", "s": u"س", "t": u"ت","u": u"و","v": u"ف", "w": u"و", "x": u"اكس",
"y": u"ي","z": u"ز" ,
"djezzy": u"جيزي","djezy": u"جيزي","djizy": u"جيزي","djizzy": u"جيزي",
"conection":u"انترنت","connection":u"انترنت","conexion":u"انترنت","connexion":u"انترنت","conx":u"انترنت","cnx":u"انترنت",
"3g": u"3ج",
```

Figure 54: La Translitération

Une synthèse des résultats obtenus après exécution de ce code est présentée dans la figure ci-dessous.

	comment	note	langue
58	الراحة djezzy هو ما هو قيود fiha	0.0	fr
78	... إذا لم يكن الأمر يتطلق بتوصيل موقعك ، فلا يتم	0.0	fr
103	Raak etchekar fi djezzy bessah zéroconnexion w...	1.0	fr
104	اتصال الصفر	1.0	fr
105	انتباه puce ooredoo خادعة ولص	1.0	fr

	comment	note	langue
58	الراحة جيزي هو ما هو قيود فيها	0.0	fr
78	... إذا لم يكن الأمر يتطلق بتوصيل موقعك ، فلا يتم	0.0	fr
103	...راك وشاكار في جيزي بصح زيرواونترنت واللو ما ع	1.0	fr
104	اتصال الصفر	1.0	fr
105	انتباه بيس اوريدو خادعة ولص	1.0	fr

Figure 55: Implémentation de la Translitération

3.1.4. Radicalisation

Nous avons utilisé dans cette phase l'API Frasa stemmer dédiée pour la langue arabe. La figure ci-dessous présente le code utilisé afin de l'implémentation.

```
from farasa.stemmer import FarasaStemmer
stemmer = FarasaStemmer(interactive=True)
Francais['comment']=Francais['comment'].apply(lambda x : stemmer.stem(x))
Francais
```

Figure 56: Code source de l'implémentation de Frasa stemmer

Une synthèse des résultats obtenus après exécution de ce code est présentée dans la figure ci-dessous.

3.1.5. Tokenisation

Dans cette partie nous avons utilisé la bibliothèque NLTK afin d'effectuer la Tokenisation.

La figure ci-dessous présente le code source utilisée pour la tokenisation.

```
import nltk
Francais['comment'] = Francais.apply(lambda row: nltk.word_tokenize(row['comment']), axis=1)
Francais
```

Figure 57: Code source de la Tokenisation

Une synthèse des résultats obtenus après exécution de ce code est présentée dans la figure ci-dessous.

	comment	note	langue
58	[راحة, جيزي, هو, ما, هو, قيد, في]	0.0	fr
78	[...إذا, لم, كان, أمر, تعلق, توصيل, موقع, لا, تم]	0.0	fr
103	[... راك, شكر, في, جيزي, صح, زيرو, إنترنت, الو]	1.0	fr
104	[اتصال, صفر]	1.0	fr
105	[انتباه, بيس, اوريدو, خادع, لص]	1.0	fr

Figure 58: Implémentation de la Tokenisation

3.1.6. Elimination des mots vides

Nous avons effectué l'élimination des mots vides à partir d'une liste d'environs 1000 mots vides algériens, la figure ci-dessous présente une partie de cette liste.

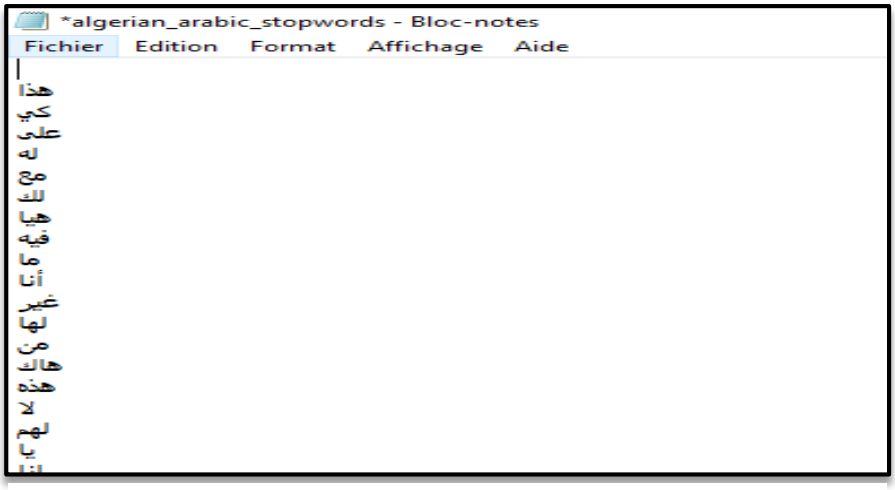


Figure 59: Liste de mots vides algériens

La figure ci-dessous présente l'élimination des mots en utilisant cette liste.

	comment	note	langue
58	[راحة, جيزي, هو, ما, هو, قيد, في]	0.0	fr
78	[...إذا, لم, كان, أمر, تعلق, توصيل, موقع, لا, تم]	0.0	fr
103	[... راك, شكر, في, جيزي, صح, زيرو, إنترنت, الو]	1.0	fr
104	[اتصال, صفر]	1.0	fr
105	[انتباه, بيس, اوريدو, خادع, لص]	1.0	fr

	comment	note	langue
58	[راحة, جيزي, قيد]	0.0	fr
78	[...أمر, تعلق, توصيل, موقع, اشتراك, جيزي, أرسل, ر]	0.0	fr
103	[...راك, شكر, جيزي, صح, زيرو, إنترنت, الو, ساعة]	1.0	fr
104	[اتصال, صفر]	1.0	fr
105	[انتباه, بيس, اوريدو, خادع, لص]	1.0	fr

Figure 60: Implémentation de l'élimination des mots vides

3.2. Entrainement des modèles

Une fois, les données prétraitées, nous passons à l'entraînement de nos modèles DZIRI-Bert, ARA-Bert, CNN-LSTM, BI-LSTM, CNN.

Le modèle DZIRI-Bert a été entrainé avec un corpus sans prétraitement puisque c'est un modèle qui gère le contenu des textes écrits en caractères arabes et latins, quant aux trois autres modèles nous les avons entrainés avec un corpus prétraité.

3.2.1. Entrainement des modèles CNN-LSTM, BI-LSTM, CNN

Dans cette partie nous allons entrainer les modèles basant sur l'apprentissage à partir de zéro.

3.2.1.1. Keras Word Embading

Nous avons utilisé le embedding de Keras pour le mappage des mots en des vecteurs de nombres réels dans un espace dimensionnel réduit grâce a quoi il est capable de savoir le contexte mot dans un document.

La figure ci-dessous (Figure 61) montre le processus du keras Word embading étape par étape.

Le processus du Word embading est comme suit :

- Vectoriser les données textuelles avec la méthode Tokenizer ()
- Appliquer une méthode de remplissage pour ajouter des zéros et définir la taille fixe dans chaque vecteur avec la méthode Texts_to_sequences ()
- Charger des données textuelles et les diviser en parties d'apprentissage et de test.

```
MAX_NB_WORDS=11000
MAX_SEQUENCE_LENGTH=25
EMBEDDING_DIM= 300
tokenizer = Tokenizer(num_words=MAX_NB_WORDS, filters='!"#$%&()*+,-./:;<=>?@[\]^_`{|}~', lower=True)
tokenizer.fit_on_texts(data['comment'].values)
word_index = tokenizer.word_index
print('Found %s unique tokens.' % len(word_index))

X = tokenizer.texts_to_sequences(data['comment'].values)
X = keras.preprocessing.sequence.pad_sequences(X, maxlen=MAX_SEQUENCE_LENGTH)
print('Shape of data tensor:', X.shape)
Y = pd.get_dummies(data['annotation']).values
print('Shape of label tensor:', Y.shape)

VOCAB_SIZE = len(tokenizer.word_index)+1
```

Figure 61: vectorisation des modèles CNN-LSTM, BI-LSTM, CNN

3.2.1.2. Initialisation des modèles CNN_Lstm, bi-lstm, cnn

Pour le modèle CNN-Lstm notre modèle sera initialisé comme suit :

```
model2= Sequential()
model2.add(Embedding(VOCAB_SIZE, EMBEDDING_DIM, input_length=X.shape[1]))
model2.add(tensorflow.keras.layers.SpatialDropout1D(0.5))
model2.add(Conv1D(filters=128, kernel_size=4, padding='same', activation='softmax'))
model2.add(MaxPooling1D(pool_size=2))
model2.add(LSTM(256, dropout=0.3, recurrent_dropout=0.2))
model2.add(Dense(3, activation='softmax'))
model2.compile(loss='categorical_crossentropy', optimizer="adam", metrics=['accuracy'])
model2.summary()
```

Figure 62: initialisation du modèle CNN-LSTM

- **MaxPooling1D :** Divise par deux la taille des entités en les sous-échantillonnant à la valeur maximale à l'intérieur d'une fenêtre, Cette couche est la raison pour laquelle un CNN peut gérer les énormes quantités de données dans les images.
- **Dropout :** Protège contre le surapprentissage en réglant aléatoirement les poids d'une partie des données à zéro.
- **Dense :** Est une couche de réseau neuronal qui est connectée en profondeur, ce qui signifie que chaque neurone de la couche dense reçoit des entrées de tous les neurones de sa couche précédente.
- **Loss :** Il s'agit d'une méthode permettant d'évaluer dans quelle mesure l'algorithme modélise l'ensemble de données. Si les prédictions sont totalement erronées, la fonction de perte produira un nombre plus élevé. S'ils sont assez bons, cela produira un nombre inférieur.
- **Optimizer :** Méthodes utilisées pour modifier les attributs du réseau de neurones tels que les poids et le taux d'apprentissage afin de réduire les pertes.

Hyperparamètre du modèle CNN-Lstm	
Input-Shape	**7044**
Dense (output)	**3**
Optimizer	**Adam**
Loss-function	**Categorical_crossentropy**

Tableau 28: Hyperparamètre du modèle CNN-Lstm

Pour le modèle Bi-lstm notre modèle sera initialisé comme suit :

```
model16= Sequential()
model16.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM, input_length=X.shape[1]))
model16.add(Bidirectional(LSTM(180, dropout=0.2, recurrent_dropout=0.5)))
model16.add(Dense(3, activation='softmax'))
model16.compile(loss='categorical_crossentropy', optimizer='rmsprop', metrics=['accuracy'])
model16.summary()
```

Figure 63: initialisation du modèle CNN-LSTM

Hyperparamètre du modèle CNN-Lstm	
Input-Shape	**7044**
Dense (output)	**3**
Optimizer	**rmsprop**
Loss-function	**Categorical_crossentropy**

Tableau 29: Hyperparamètre du modèle CNN-Lstm

Pour le modèle Bi-lstm notre modèle sera initialisé comme suit :

```
md7= Sequential()
md7.add(Embedding(VOCAB_SIZE, EMBEDDING_DIM, input_length=X.shape[1]))
md7.add(Conv1D(filters=100, kernel_size=4, padding='same', activation='relu'))
md7.add(MaxPooling1D(pool_size=2))
md7.add(Dropout(0.5))
md7.add(Conv1D(filters=32, kernel_size=4, padding='same', activation='relu'))
md7.add(MaxPooling1D(pool_size=2))
md7.add(Flatten())
md7.add(Dense(64, activation='relu'))
md7.add(Dropout(0.3))
md7.add(Dense(3, activation='sigmoid'))
md7.compile(loss='categorical_crossentropy', optimizer='rmsprop', metrics=['accuracy'])
```

Figure 64: initialisation du modèle CNN-LSTM

Hyperparamètre du modèle CNN-Lstm	
Input-Shape	**7044**
Dense (output)	**3**
Optimizer	**rmsprop**
Loss-function	**Categorical_crossentropy**

Tableau 30: Hyperparamètre du modèle CNN

3.2.1.3. Résultats obtenus des modèles CNN-LSTM, BI-LSTM, CNN

Dans cette section nous allons présenter les résultats obtenus après formation des modèles CNN-LSTM, BI-LSTM, CNN.

- **Résultats obtenus par le modèle CNN-LSTM :**

En premier, nous avons utilisé le model CNN-LSTM sur notre Dataset.

La figure ci-dessous montre les résultats obtenus du modèle CNN-LSTM :

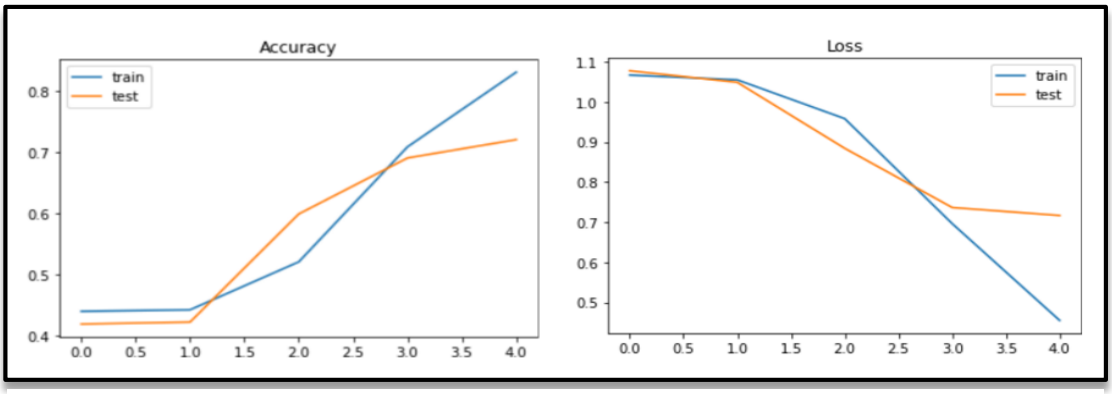


Figure 66: Accuracy et loss pour le model CNN-LSTM.

D'après la figure on remarque que la précision de l'apprentissage augmente avec le nombre d'époque, ce qui reflète qu'à chaque époque le modèle apprend de plus en plus. La même chose pour l'erreur, l'erreur d'apprentissage diminue, d'autre part la validation augment proportionnellement avec le nombre d'époques.

Le modèle a donné une valeur Accuracy de 85% avec une valeur loss de 39% et val-Accuracy de 75% avec un Val_Loss de 59%.

- **Résultats obtenus par le modèle Bi-LSTM :**

La figure ci-dessous montre les résultats obtenus du modèle Bi-lstm.

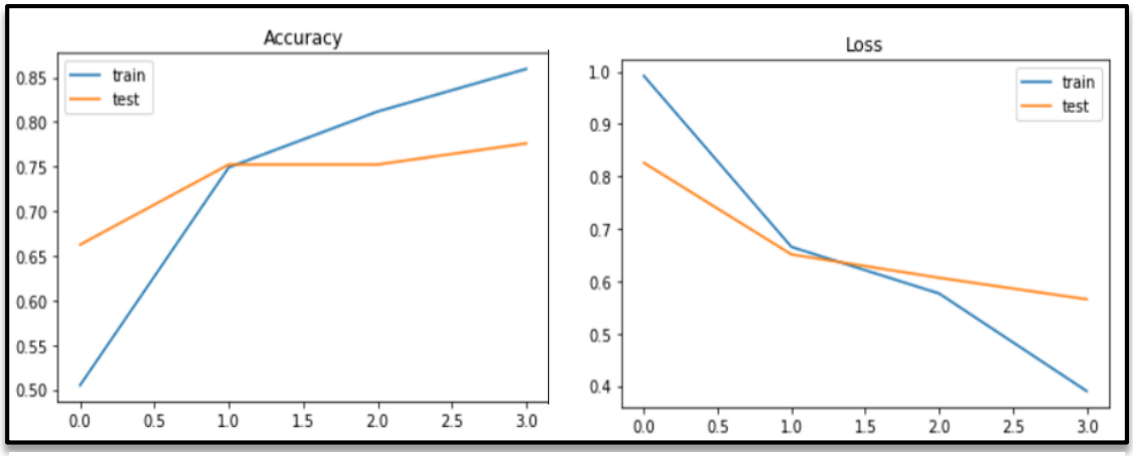


Figure 67: Accuracy et loss pour le model Bi-lstm.

D'après la figure on remarque aussi que la précision de l'apprentissage augmente avec le nombre d'époques. Elle se stabilise un certain moment et après elle commence a

augmenté à nouveau pour la validation.

La même chose pour l'erreur, l'erreur d'apprentissage diminue, d'autre part la validation augment avec le nombre d'époque.

Le modèle a donné une valeur Accuracy de 85% avec une valeur loss de 38% et val-Accuracy de 77% avec un Val_Loss de 56%.

- **Résultats obtenus par le modèle CNN :**

La figure ci-dessous montre les résultats obtenus du modèle CNN :

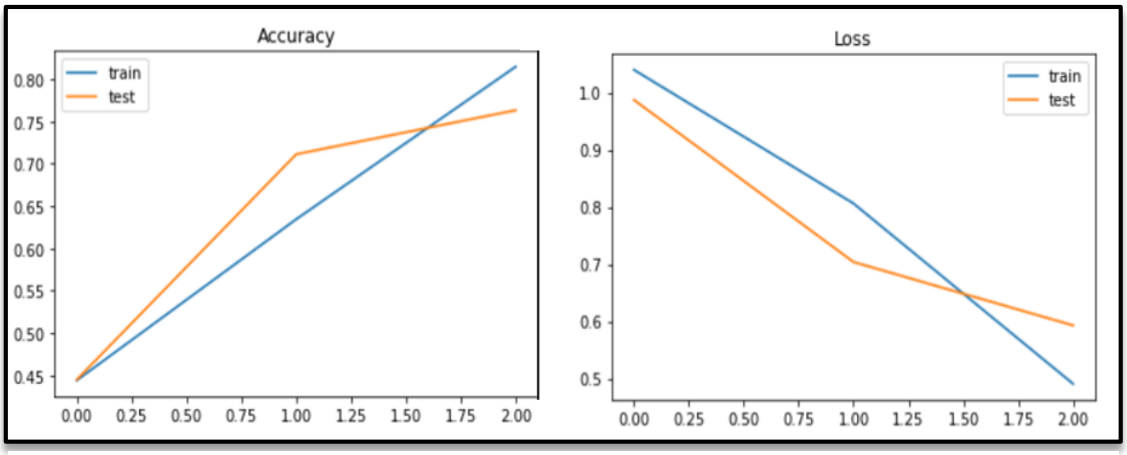


Figure 68 : Accuracy et loss pour le model CNN.

D'après la figure, nous remarquons que la précision de l'apprentissage augmente avec le nombre d'époques, l'entrainement et la validation se croisent dans la 2 éme époque.

La même chose pour l'erreur, l'erreur d'apprentissage diminue et d'autre part la validation augment avec le nombre d'époque.

Le modèle a donné une valeur Accuracy de 81% avec une valeur loss de 49% et val-Accuracy de 77% avec un Val_Loss de 59 %.

3.2.1.3. Comparaison des modèles CNN_LSTM, BI-LSTM, CNN

Dans la table ci-dessous, nous allons comparer les trois modèles se basant sur l'apprentissage à partir de zéro en utilisant les métriques d'évaluation suivantes : **loss, Accuracy, validation_loss et validation_accuracy.**

Modèle	Loss	Accuracy	Val_Loss	Val_Accuracy
CNN-LSTM	39%	85%	59%	75%
Bi-lstm	38%	85%	59%	76%
CNN	49%	81%	81%	77%

Tableau 31: table comparative des résultats des modèles.

Nous remarquons d'après les résultats obtenus dans le tableau ci-dessus que le meilleur modèle est celui de CNN avec un résultat de Val_Accuracy égale à 78% avec un taux

d'erreur de 58 %. Ce n'est pas vraiment un bon résultat et ceci est dû aux conditions suivantes :

- Les données ne sont pas structurées.
- La taille des données est très petite sachant que ces modèles apprennent à partir de zéro.

3.2.2. Entrainement des modèles DZIRI-Bert et ARA-Bert

Dans cette section nous allons entrainer les modèles basant sur l'apprentissage par transfert DZIRI-Bert et ARA_bert.

3.2.2.1. Initialisation des modèles DZIRI-Bert et ARA-Bert

Pour les modèles DZIRI-Bert notre modèle sera initialisé comme suit :

Hyperparamètre du modèle DZIRI-Bert		Valeurs
Taille du vocabulaire		50000
Longueur maximale des commentaires		180
Fine tuning	Learning rate	2e-5
	Batch size train	8
	Batch size val	4

Tableau 32: Hyperparamètre du modèle DZIRI-Bert

- **Fine tuning** : prend un modèle qui a déjà été formé pour une tâche particulière, puis le peaufine pour lui faire effectuer une deuxième tâche similaire. Par exemple, un réseau d'apprentissage en profondeur qui a été utilisé pour reconnaître les voitures peut être affiné pour reconnaître les camions.
- **Learning rate** : le taux d'apprentissage est un paramètre de réglage dans un algorithme d'optimisation qui détermine la taille du pas à chaque itération tout en se déplaçant vers un minimum d'une fonction de perte.

Pour les modèles Ara-Bert notre modèle sera initialisé comme suit :

Hyperparamètre du modèle Ara-Bert		Valeurs
Taille du vocabulaire		64000
Longueur maximale des commentaires		180
Fine tuning	Learning rate	2e-5
	Batch size train	8
	Batch size val	4

Tableau 33: Hyperparamètre du modèle Ara-Bert

3.2.2.2. Résultats obtenus des modèles DZIRI_Bert et ARA-Bert

Dans cette section, nous allons présenter les résultats obtenus après formation des modèles DZIRI-Bert et ARA-Bert.

Dans le tableau ci-dessous, résume ces résultats :

	DZIRI-Bert	ARA-Bert
Accuracy	83%	81%
F1	82%	80%
Précision	82%	80%
Recall	82%	80%

Tableau 34: Résultats d'entrainement des modelés BERT

Nous remarquons d'après les résultats obtenus dans le tableau ci-dessous que le meilleur modèle est celui de DZIRI-bert avec une valeur d'Accuracy égale à 83% avec valeurs de F1, Précision, Recall de 82 %. Nous jugeons ce résultat comme étant logique car le modèle DZIRI-Bert est fait spécialement pour traiter le dialecte algérien.

Dans les figures suivantes, nous allons montrer les résultats obtenus de façon plus détaillée à travers des graphiques. Les graphiques suivants nous permettent de suivre l'évolution des métriques Accuracy, F1, précision, Recall.

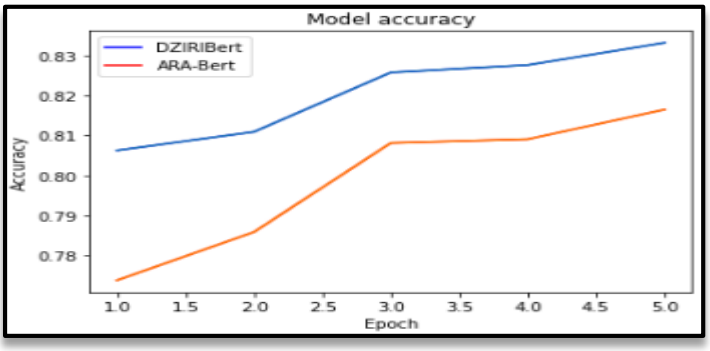


Figure 69: Evolution d'Accuracy des modèles DziriBert et ARA-Bert

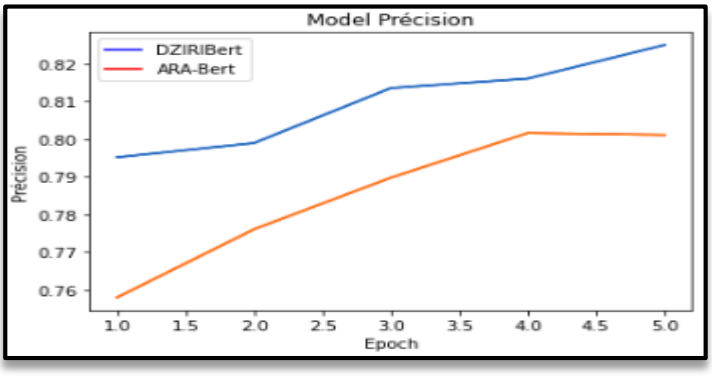


Figure 70: Evolution de précision des modèles DZIRI-Bert et ARA-Bert

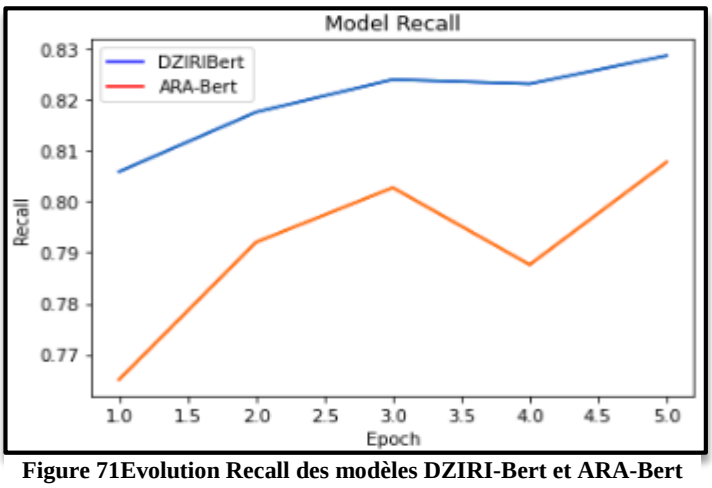


Figure 71Evolution Recall des modèles DZIRI-Bert et ARA-Bert

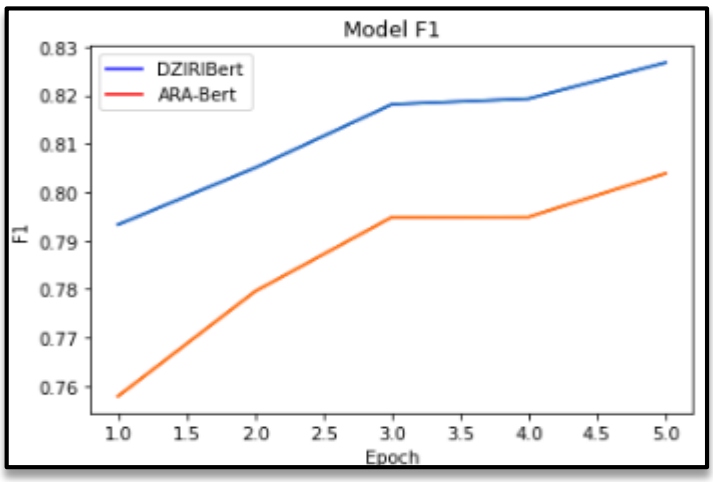


Figure 72: Evolution F1 des modèles DZIRI-Bert et ARA-Bert

3.3. La comparaison des modèles

La table ci-dessous représente tous les résultats des modèles suivants : CNN_Lstm, Bi-Lstm, CNN, DZIRI-Bert, ARA-Bert.

Modèle	Précision	Recall	F1_Score	Accuracy
CNN_Lstm	73%	64%	70%	75%
Bi-Lstm	79%	70%	72%	76%
CNN	80%	73%	73%	77%
DZIRI-Bert	82%	82%	82%	83%
ARA-Bert	80%	80%	80%	81%

Tableau 35: Tableau 31 : résultat de l'ensemble de test des différents modèles.

Tous d'abord, nous remarquons dans ce tableau que les modèles qui se basent sur l'apprentissage par transfert (DZIRI-Bert, ARA-Bert) ont donné des meilleurs résultats par rapport aux modèles (CNN_Lstm, Bi-Lstm, CNN), sachant que le

meilleur modèle est celui de DZIRI-Bert avec un taux d'Accuracy excellent qui est de 83% et c'est un résultat qui est logique car ce modèle est fait pour traiter spécialement le Dialecte Algérien.

3.3. Visualisation graphique

Nous avons effectué la classification avec le modèle qui a obtenu les meilleurs résultats qui est DZIRI-Bert. Cela a été fait sur un échantillon de Feedback clients de l'entreprise Djezzy. Après nettoyage de ce dernier qui a consisté à la suppression des caractères non appartenant au code américain normalisé pour l'échange d'information (ASCII) et des commentaires doublons nous avons obtenu 774 commentaires auxquels nous avons réalisé une visualisation graphique qui sera détaillé ci- dessous.

Nous allons présenter ci-dessous les visualisations graphiques que nous avons réalisées dans le but de mieux visualiser les résultats et présenter notre projet.

3.3.1. Interface graphique

Tout d'abord l'utilisateur doit saisir son texte à la fin de visualiser les résultats et valider avec le bouton, par la suite il reçoit un résultat de la langue détecté ainsi que du prétraitement effectué sur le texte saisi qui sont : stemming, tokenisation et aussi la polarité de ce dernier.

la figure suivante présente le diagramme de séquence pour l'utilisation de cette interface.

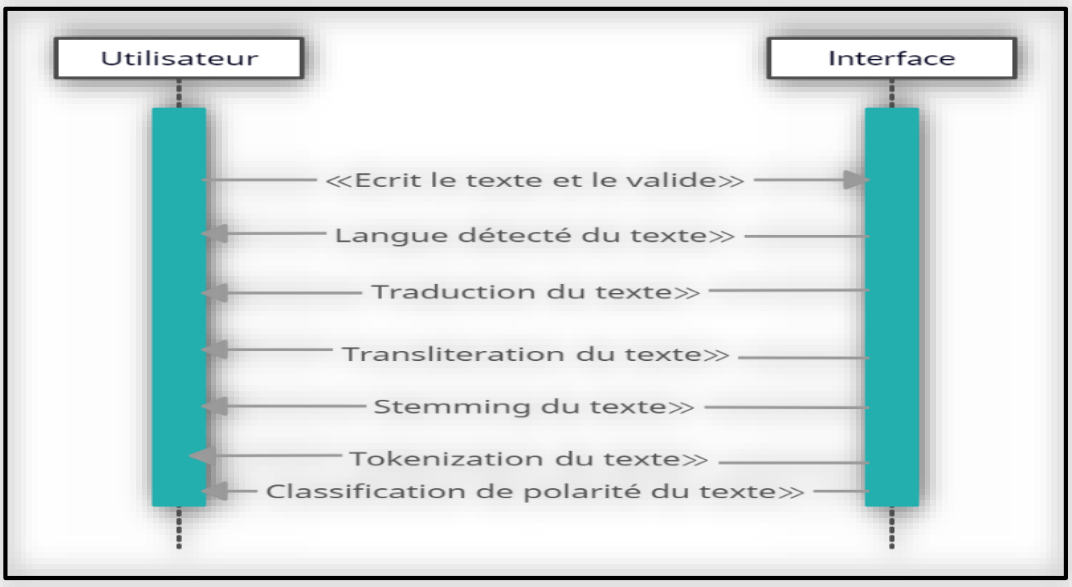


Figure 73: Diagramme de séquence de l'utilisation de l'interface

Comme montré ci-dessous, cette interface nous permet d'écrire un texte en dialecte algérien et de visualiser les étapes de prétraitement ainsi que la classification du sentiment de ce dernier.

Comme expliqué précédemment nous avons commencé notre prétraitement par la détection de la langue, pour le cas de cet exemple du texte « offre mliha mrc » a été détecté comme texte non arabe et non français vu la présence du terme « mliha » qui n'appartient pas au dictionnaire français. L'étape suivante est la traduction, dans notre cas seulement le mot « offre » est en français et ce qui explique la traduction de ce mot uniquement. Par la suite, l'étape de la translitération des termes non traduits ainsi du terme « mrc » en langage sms et enfin le stemming avec l'outil nlp arabe Frasa stemmer et la tokenisation. Bien évidement la classification du texte en positif ce qui prouve l'efficacité du modèle qu'on a réalisé.

4. Power BI

Nous avons réalisé un tableau de bord afin d'aider l'entreprise Djezzy à suivre avec efficacité les services qu'elle propose ainsi que le niveau de satisfaction de ses clients. C'est également un moyen de savoir leurs attentes par rapport aux services proposés et futurs.

Dans cette partie de visualisation de données sur Power BI, nous avons représenté nos résultats sous forme de cercle graphique avec des pourcentages de chaque catégorie de sentiment, sous forme de distribution des sentiments par mois et sous forme de nuage de mots clés afin d'analyser les mots les plus utilisés ou populaires dans nos données de classification. Aussi nous avons effectué le calcul du score NPS qui est un indicateur permettant de connaître et de mesurer la satisfaction et la fidélisation des clients. Son objectif est d'identifier les trois types de clients : passifs, promoteurs et détracteurs. Nous l'avons calculé dans le but d'assister le service Customer care de l'entreprise Djezzy à mieux diriger ses stratégies.

La figura ci-dessus (Figure 67) représente analyse sur les avis de client de djezzy avec Power BI.

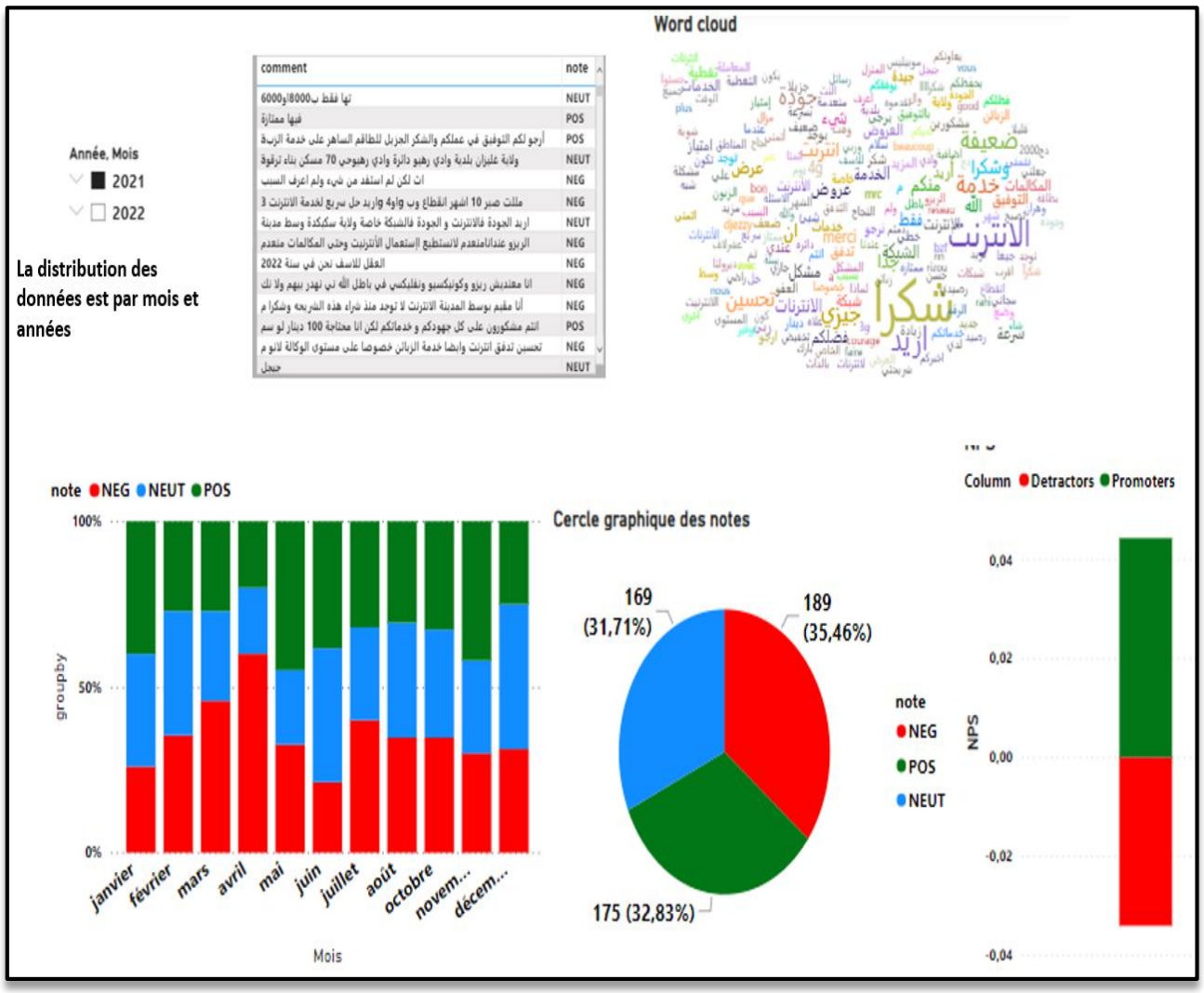


Figure 74: Analyse sur les avis de client de djezzy avec Power BI

5. Conclusion

Dans ce dernier chapitre, nous avons présenté et discuter les différents résultats que nous avons obtenus avec différentes approches sachant que nous avons exploité six modèles de Deep Learning ainsi que l'interface finale et nous avons fait l'analyse des données. Les résultats obtenus sont satisfaisants et démontre qu'on peut trouver des solutions admissibles pour le problème de la complexité de l'analyse de sentiments pour le dialecte algérien.

Conclusion générale

1. Conclusion

L'analyse des sentiments est devenue un domaine de recherche très populaire. Mais il existe encore de nombreux problèmes face à ce domaine, car l'analyse des sentiments traite des données non structurées basées sur un texte en plus de ça ce domaine n'est pas très adapté pour notre société qui a une très grande variété dialectale. Ces derniers temps une pléthore de modèles ont été proposés et ils ont prouvé leurs performances en donnant des bons résultats. Le succès de ces modèles est attribué à leur capacité d'apprentissage automatique.

Le travail réalisé s'inscrit dans le domaine d'analyse des sentiments des Feedbacks des Clients Djezzy exprimés en dialecte Algérien. Nous nous sommes particulièrement concentrés sur la multi classification des sentiments en dialecte Algérien en proposant des modèles se basant sur l'apprentissage en profondeur.

En premier lieu dans ce mémoire Nous avons présentés l'entreprise Djezzy en plus nous avons présentés l'analyse des sentiments, y compris ses domaines d'utilisation, ses niveaux, ses défis et les types de classification ainsi que les approches qui sont adoptées dans la littérature. Ensuite nous avons présenté la langue arabe, en commençant par présenter ses défis, sa complexité et ses dialectes. Nous avons également abordé le dialecte algérien et sa complexité et nous avons aussi discuté les notions fondamentales des approches de Deep Learning qui ont été utilisées dans notre cas, leurs fonctionnements et les notions générales sur les réseaux de neurones avant de parler des études déjà faits.

Ensuite, nous avons abordé les travaux antérieurs dans le contexte de l'Analyse de Sentiments et les dialectes arabes et nous avons vu les différentes techniques déjà existantes qui ont été implémentées ainsi que leurs résultats et leurs configurations variées.

Par la suite, nous avons fait la modélisation de notre système de classification passant par plusieurs étapes de prétraitement. Nous avons utilisé des modèles de deux types d'approches d'apprentissage en profondeur : ceux de l'apprentissage à partir de zéro qui sont ; CNN_Lstm, Bi-Lstm, CNN et d'autres se basant sur l'apprentissage par transfert qui sont DZIRI-Bert et Ara-Bert.

Nous avons entrainé nos modèles sur des données non structurées de 6743 commentaires. Les résultats que nous avons obtenus avec les modèles qui se basent sur l'apprentissage par transfert ont été nettement meilleurs que les autres modèles.

Sachant que le meilleur modèle était celui de DZIRI-Bert avec un taux d'Accuracy égale à 83% et qui est plutôt tr-s encourageant dans notre cas d'étude.

Se basant sur ces résultats, nous avons développé une interface pour simuler les différentes étapes de traitement des commentaires, allant de leur prétraitement jusqu'à leur classification et nous avons aussi réalisé un tableau de bord récapitulatif des données de l'entreprise selon diverses périodes temporelles en utilisant le logiciel power BI.

2. Perspectives

Bien que les résultats obtenus -dans cette première version- ayant été jugés comme étant encourageants par l'entreprise et que les objectifs initiaux ont été atteints, il persiste toutefois quelques perspectives d'amélioration afin d'enrichir notre travail et espérer atteindre de meilleurs résultats. Ces perspectives se résument en :

- Augmenter le nombre de données pour minimiser le taux d'erreur. En effet, il serait judicieux de considérer la collecte sur une période de temps plus étendue et espérer ainsi avoir un Dataset plus étoffé pour les tets.
- Améliorer les résultats obtenus en testant d'autres techniques et modèles d'apprentissage profond.
- Utiliser d'autres techniques de représentation de donnée et plus évoluées du Word Embedding comme le word2vec et le modèle Glove.
- La création d'un modèle de traduction automatique personnalisé plutôt que l'adoption de celui de Google. En effet, le domaine Télécom étant particulier en plus des spécificités du dialecte Algérien, la réflexion à un modèle de traduction contextualisé pourrait s'avérer intéressante.

Références bibliographiques

Références bibliographiques

[1] El-Din, D. M. (2016, Juin). Sentiment Analysis of Online Papers (SAOOP). Master Thesis of sentiment Analysis. Cairo, Egypt: Cairo University.

[2] Dang, N. C., Moreno-García, M. N., & Prieta, F. D. (2020, juin 5). Sentiment Analysis Based on Deep Learning: A Comparative Study. *Electronics, 9 (3), 483*, p. 29.

[3] DanielGraziotinb, V.Mäntyläa, M., & MiikkaKuutilaa. (2018, fevrier). The evolution of sentiment analysis. *Computer Science Review*, pp. 16-32.

[4] Lohard, A., & Boullier, D. (2012). OPINION MINING ET SENTIMENT ANALYSIS. OpenEdition Press.

[5] Zulfadzli Drus, Haliyana Khalid, Sentiment Analysis in social media and Its Application: Systematic Literature Review, Procedia Computer Science, Volume 161, 2019, Pages 707-714, ISSN 1877-0509.

[6] Pawar, Kishori & Shrishrimal, P & Deshmukh, Ratnadeep. (2015). Twitter Sentiment Analysis: A Review. IJSER. 6.

[7] Kharde, V. A., & Sonawane, S. (2016). Sentiment Analysis of Twitter Data: A Survey of techniques. International Journal of Computer Applications, 139(11), (0975 – 8887).

[8] Varghese, R., & M, J. (2013). A SURVEY ON SENTIMENT ANALYSIS AND OPINION MINING. *IJRET: International Journal of Research in Engineering and Technology*, 313.

[9] Pang, Bo ; Lee, Lillian (2008). "4.1.2 Détection de la subjectivité et identification d'opinion ». Exploration d'opinions et analyse des sentiments. Now Publishers Inc.

[10] Mehta, P., & Pandya, D. (2020). A Review On Sentiment Analysis Methodologies, INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH,602.

[11] Ghorbani, A. A., & Karamibekr, M. (2013). Sentence Subjectivity Analysis in Social Domains. Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). Fredericton.

[12] Marzak, A., Benlahmar, E. H., & Lamrani, E.-k. (2015). OPINION MINING : UN ÉTAT DE L'ART. La troisième journée sur les Technologies d'Information et de Modélisation TIM'15. Casablanca.

[13] Liu, B. (2012). Sentiment Analysis and Opinion Mining. Chicago: Morgan & Claypool , p25

[14] Badia, K., & Benslimane, S. M. (2019). Multilingual sentiments analysis to improve the quality of services provided by Algerian telephone operators. Third Edition of the National Study Day on Research on Computer Sciences: JERI'2019At, (p. 3). Saida.

[15] MAHAMMED, F. A. (2018). APPROCHES D'APPRENTISSAGE AUTOMATIQUE POUR LA DÉTECTION DU SPAM WEB : EXPLORATION DE DIVERSES CARACTÉRISTIQUES. MONTRÉAL : UNIVERSITÉ DU QUÉBEC.

[16] Astya, P., & Shahnawaz*. (2017). Sentiment Analysis: Approaches and Open issues. International Conference on Computing, Communication and Automation, (p. 10). Greater Noida

[17] A. Hadhémi, A. Ferchichi, E. Souissi et J. Younes, «Un état de l'art du traitement automatique du dialecte tunisien,» TAL Traitement Automatique des Langues, vol. 59, n° 13, pp. 93-117, 2019.

[18] Abir Masmoudi Dammak, Approche hybride pour la reconnaissance automatique de la parole en langue arabe, Université du Maine, 2016.

[19] Farag Dardour, Langue enseignée et dialecte arabe: quelle méthodologie et quelle formation pour l'acquisition communicative en arabe standard, Université Nancy 2, 2008.

[20] LECLERC, Jacques. « Une appellation controversée » La famille afro-asiatique, Québec, CEFAN, Université Laval, 29 nov. 2020, [https://www.axl.cefan.ulaval.ca/monde/famarabe.htm], (24 mai 2022).

[21] A. Farghaly et K. Shaalan, «Arabic Natural Language Processing: Challenges and Solutions,» ACM Transactions on Asian Language Information Processing, vol. 8, n° %14, 2009.

[22] N. Ayoubi, «middleeasteye,» [En ligne]. Available: https://www.middleeasteye.net/discover/five-major-spoken-arabic-dialects-unique. [Accès le 21 février 2022].

[23] A. A. Team, «Asian Absolute,» [En ligne]. Available: https://asianabsolute.co.uk/blog/2016/01/19/arabic-language-dialects/. [Accès le 19 janvier 2016].

[24] Guellil, Imane & Azouaou, Faical & Saadane, Houda & Semmar, Nasredine. (2018). Une approche fondée sur les lexiques d'analyse de sentiments du dialecte algérien. TAL Traitement Automatique des Langues. 58. 41 à 65.

[25] Imène Guellil, Faiçal Azouaou. ASDA : Analyseur Syntaxique du Dialecte Algérien dans un but d'analyse sémantique. RFIA-RJCIA 2016, Association Française pour l'Intelligence Artificielle, Jun 2016, Clermont Ferrand, France.

[26] Salima Brachemi-Meftah, Fatiha Barigou. Algerian Dialect Sentiment Analysis : State of Art, in 2020 21st International Arab Conference on information Technology (ACIT), Novembre 2020, p. 1-7, doit : 10.1109/ACIT50332.2020.9300060

Delizarch, C. (2021, février 07). Deep Learning : qu'est-ce que c'est ?

[27] Nhan Cach Dang, M. N.-G. (2021, Mars 14). Sentiment Analysis Based on Deep Learning: A Comparative Study. Récupéré sur Electronics: https://www.mdpi.com/2079-9292/9/3/483

[28] Madhaven, S. (2019). What is the difference between deep learning and transfer learning? Récupéré sur Quora: https://www.quora.com/Whats-the-difference-between-reinforcement-Learning-and-Deep-learning.

[29] Truong, P. H. (2016, juin). Optimisation des performances de la machine synchrone à réluctance variable : approches par la conception et par la commande. Vietnam, Vietnam National University, Ho Chi Minh City, chine.

[30] *Réseaux de Neurones. (2020). Récupéré sur Statistic: https://www.statsoft.fr/concepts-statistiques/reseaux-de-neurones-automatises/reseaux-de-neurones-automatises.php*

[31] Lin, D. C.-E. (2021, juin). *8 Simple Techniques to Prevent Overfitting*. Récupéré sur Medium: https://towardsdatascience.com/8-simple-techniques-to-prevent-overfitting-4d443da2ef7d

[32] Lin, D. C.-E. (2021, juin). *8 Simple Techniques to Prevent Overfitting*. Récupéré sur Medium: https://towardsdatascience.com/8-simple-techniques-to-prevent-overfitting-4d443da2ef7d

[33] 1D Convolution. (2021, septembre). Récupéré sur Knowledge Center: https://peltarion.com/knowledge-center/modeling-view/build-an-ai-model/blocks/1d-convolution

[34] sharma, P. (2021, 02 01). *Keras LSTM Layer Explained for Beginners with Example*. Récupéré sur Making AI simple: https://machinelearningknowledge.ai/keras-lstm-layer-explained-for-beginners-with-example/#:~:text=Long%20Short%2DTerm%20Memory%20Network,feedback%20loop%20in%20its%20architecture.

[35] Rebeen Ali Hamad, W. L. (2020, juin). Joint Learning of Temporal Models to Handle Imbalanced Data for Human Activity Recognition. *Applied Sciences* , p. 10(15):5293.

[36] Hover, R. (2018, Novembre 10). BERT Explained: State of the art language model for NLP. Récupéré sur Medium: https://towardsdatascience.com/deep-learning-techniques-for-text-classification-78d9dc40bf7c

[37] Word Embedding Example with Keras in Python. (2019, 05 22). Récupéré sur Data Tech note: https://www.datatechnotes.com/2019/05/word-embedding-with-keras-in-python.html

[38] Difference between Loss, Accuracy, Validation loss, Validation accuracy in Keras. (2022, Aout 11). Récupéré sur Code Monc: https://www.javacodemonk.com/difference-between-loss-accuracy-validation-loss-validation-accuracy-in-keras-ff358faa