



UNIVERSITÉ CADI AYYAD

ÉCOLE SUPÉRIEURE DE TECHNOLOGIE - SAFI

DÉPARTEMENT: INFORMATIQUE

RAPPORT: PROJET DE FIN D'ÉTUDE

Deep Learning: L'analyse des sentiments des tweets arabes

Réalisé par :
ELKAROUAOUI Ali
BELAHRECH Abderahmanne
BADRI Mohammed Amine

Encadré par :
Mr. ELHOURRI Soufiane
ELKASSAH Wiam

ANNÉE UNIVERSITAIRE : 2021/2022

Dédicace

Je dédie ce projet à nos chers parents, sœurs, frères, amis et à tous les enseignants qui nous ont aidés dans ce projet. Pour leur patience, leurs encouragements et leur engagement.

Remerciements

C'est avec un immense plaisir que je réserve ces quelques lignes en signe de gratitude et de reconnaissance à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

Notre très chers remerciements vont à notre encadrent Dr. ELHOURRI Soufiane pour son encadrement si particulier et tutrice ELKASSAH Wiam qui nous a aider d'accomplir ce projet.

Enfin, j'exprime toute ma reconnaissance à mon école : l'Ecole Supérieur de Technologie Safi, en personne de Monsieur le Directeur, ainsi qu'à tous les enseignants et les membres de jury.

Sommaire

| | |
|--|-----------|
| List of Figures | 8 |
| List of tables | 9 |
| 1 Introduction Générale | 12 |
| 1.1 Contexte globale : | 14 |
| 1.2 Problématique : | 14 |
| 1.3 Contributions: | 14 |
| 1.4 Organisation du rapport : | 15 |
| 1.5 Cahier de charge : | 15 |
| 1.5.1 Aperçu du projet | 15 |
| 1.5.1.1 Resume du projet | 15 |
| 1.5.1.2 Buts et objectifs du projet et résultats opérationnels | 15 |
| 1.5.1.3 Risques et contraintes du projet | 16 |
| 1.5.1.4 Produits livrables | 16 |
| 1.5.1.5 Jalons et planning prévisionnels | 16 |
| 1.5.2 Organisation du projet | 18 |
| 1.5.2.1 Structure de l'équipe de projet | 18 |
| 1.5.2.2 Rôles et responsabilités | 18 |
| 1.5.3 Spécifications fonctionnelles | 18 |
| 1.5.4 Spécifications techniques | 19 |
| 2 Partie théorique | 20 |
| 2.1 Intelligence Artificielle | 22 |
| 2.2 Machine learning | 22 |

| | | |
|----------|--|-----------|
| 2.2.1 | Type d'apprentissage automatique: | 22 |
| 2.2.2 | Algorithmes d'apprentissage automatique: | 23 |
| 2.3 | Deep learning | 24 |
| 2.3.1 | Réseaux neurones | 24 |
| 2.3.2 | Le modèle du perceptron | 25 |
| 2.3.3 | MLP | 26 |
| 2.3.4 | Les réseaux de neurones profonds | 27 |
| 2.3.4.1 | L'intérêt des architectures profondes | 27 |
| 2.3.4.2 | Réseaux de neurones convolutifs | 28 |
| 2.3.4.3 | Réseaux de neurones récurrents | 28 |
| 2.4 | L'analyse des sentiments | 29 |
| 2.4.1 | Définition | 29 |
| 2.4.2 | Fonctionnement de l'analyse des sentiments | 30 |
| 2.4.3 | Les avantages de l'analyse des sentiments | 30 |
| 2.4.4 | L'Analyse des Sentiments sur les Réseaux Sociaux | 31 |
| 2.4.5 | L'Analyse des Sentiments avec Twitter | 32 |
| 2.5 | NLP | 32 |
| 2.5.1 | Nettoyage des données | 33 |
| 2.5.2 | Normalisation | 33 |
| 2.5.3 | Stemming | 34 |
| 2.5.4 | Lemmatisation | 35 |
| 2.5.5 | Tokenisation | 35 |
| 2.5.6 | Vectorisation | 35 |
| 2.6 | Conclusion | 36 |
| 3 | Realisation | 37 |
| 3.1 | Objectives | 39 |
| 3.2 | Outils | 39 |
| 3.2.1 | Language de programmation utilisé (Python) | 39 |
| 3.2.2 | L'IDE utilisé (Pycharm) | 40 |
| 3.2.3 | Google Colab | 41 |

| | | |
|---------|--|----|
| 3.2.4 | Scikit-learn | 41 |
| 3.2.5 | Pandas | 42 |
| 3.2.6 | Numpy | 42 |
| 3.2.7 | Tensorflow | 43 |
| 3.2.8 | Keras | 43 |
| 3.2.9 | Matplotlib | 44 |
| 3.2.10 | Re | 44 |
| 3.2.11 | NLTK | 44 |
| 3.2.12 | Streamlit | 44 |
| 3.3 | Application | 45 |
| 3.3.1 | Dataset | 45 |
| 3.3.1.1 | Echantillon du dataset | 45 |
| 3.3.1.2 | Nombre des tweets negatives et positives | 46 |
| 3.3.2 | Pretraitement des donnees | 46 |
| 3.3.3 | Resultat des models | 48 |
| 3.3.4 | Application web | 49 |
| 3.4 | Conclusion | 50 |
| 4 | Conclusion | 51 |
| | Bibographie | 52 |

Liste des Figures

| | | |
|------|---|----|
| 1.1 | Jalon du projet | 17 |
| 1.2 | Organnigramme presentant l'equipe | 18 |
| 2.1 | Schéma qui montre un perceptron simple | 26 |
| 2.2 | Schema qui montre une représentation graphique d'un MLP | 26 |
| 2.3 | Schema qui montre une représentation graphique d'un CNN | 28 |
| 2.4 | Schema qui montre une représentation graphique d'un RNN | 29 |
| 2.5 | L'analyse des sentiments (Positive,Neutre,Negative) | 30 |
| 2.6 | Logo de Twitter | 32 |
| 2.7 | Norrnormalisation des tweets | 34 |
| 3.1 | Logo de python | 40 |
| 3.2 | Logo de pycharm | 40 |
| 3.3 | Logo de colab | 41 |
| 3.4 | Logo de Sklearn | 41 |
| 3.5 | Logo de Pandas | 42 |
| 3.6 | Logo de Numpy | 42 |
| 3.7 | Logo de Tensorflow | 43 |
| 3.8 | Logo de Keras | 43 |
| 3.9 | Logo de Matplotlib | 44 |
| 3.10 | Logo de Streamlit | 45 |
| 3.11 | Cinq premiers cellules du dataset | 45 |
| 3.12 | Code pour realiser les plots suivants | 46 |
| 3.13 | Nombre des tweets negatives et positives | 46 |
| 3.14 | Dataset avant pretraitement | 47 |

| | |
|---|----|
| 3.15 Dataset apres pretraitement | 47 |
| 3.16 Resultat du MLP et Logistic regression | 48 |
| 3.17 Resultat du Random Forest et Naive Bayes | 48 |
| 3.18 Resultat du SVC | 49 |
| 3.19 Resultat du CNN | 49 |
| 3.20 L'application web | 49 |

Liste des Tableaux

| | | |
|-----|---|----|
| 1.1 | Buts et objectifs du projet | 16 |
| 1.2 | Jalons et planning prévisionnel | 17 |
| 1.3 | Rôles et responsabilités | 19 |
| 1.4 | Spécifications techniques | 19 |

Resume

Nous présentons une méthode d'analyse de sentiments à partir de messages issus de l'application Twitter en utilisant **Deep Learning**.

Nous utilisant **CNN** comme modèle pour classifier les tweets. Nous présentons finalement une application web permettant de classifier des tweets arabes.

Abstract

We present a method of analyzing feelings from messages from the Twitter app using **Deep Learning**. We use **CNN** as a model to classify tweets. Finally, we present a web application to classify arabic tweets.

1

Introduction Générale

Sommaire

| | |
|--|----|
| 1.1 Contexte globale : | 14 |
| 1.2 Problématique : | 14 |
| 1.3 Contributions: | 14 |
| 1.4 Organisation du rapport : | 15 |
| 1.5 Cahier de charge : | 15 |
| 1.5.1 Aperçu du projet | 15 |
| 1.5.1.1 Resume du projet | 15 |
| 1.5.1.2 Buts et objectifs du projet et résultats opérationnels | 15 |
| 1.5.1.3 Risques et contraintes du projet | 16 |
| 1.5.1.4 Produits livrables | 16 |
| 1.5.1.5 Jalons et planning prévisionnels | 16 |
| 1.5.2 Organisation du projet | 18 |
| 1.5.2.1 Structure de l'équipe de projet | 18 |
| 1.5.2.2 Rôles et responsabilités | 18 |

| | | |
|-------|---|----|
| 1.5.3 | Spécifications fonctionnelles | 18 |
|-------|---|----|

| | | |
|-------|-------------------------------------|----|
| 1.5.4 | Spécifications techniques | 19 |
|-------|-------------------------------------|----|

1.1 Contexte globale :

Dans le cadre de notre scolarité à l'École Supérieure de technologie de Safi, on a été amené à réaliser un projet de fin d'études (PFE). Ce projet avait pour but de créer une application d'analyse des sentiments. contenus dans les tweets arabes en utilisant les techniques du Deep learning.

1.2 Problématique :

L'analyse de sentiment est une tâche de traitement automatique des langues et d'extraction d'information. Dans le cas d'un texte donné, la polarité du texte doit être identifiée comme étant soit positive ou négatif. Il y a plusieurs méthodes, références de performance et ressources pour réaliser cette tâche. La polarité d'un sentiment peut être calculée selon plusieurs seuils et peut-être vus comme plusieurs différentes classes. Dans notre projet, nous considérons que les textes ne font partie que de deux catégories (classification binaire): soit le texte est positif, soit le texte est négatif.

Récemment, l'analyse de sentiments a reçu beaucoup d'attention non seulement de la part de la recherche scientifique mais aussi par les domaines de la publicité et du marketing. Cela peut être attribué aux récentes avancées dans les réseaux sociaux et à la rapidité du relais de l'information.

Les grandes masses de données réelles issues des réseaux sociaux sont largement utilisées pour, justement, l'analyse des sentiments. Analyser les messages récents issus des réseaux sociaux pourrait donner l'opinion générale des utilisateurs envers un sujet spécifique.

1.3 Contributions:

Lors de notre projet de fin d'études nous avons réalisé une application web permettant de :

- Prétraiter et nettoyer le texte.
- Former un modèle d'apprentissage profond.

- Extraire les sentiments des tweets.
- Effectuer une prédiction.

1.4 Organisation du rapport :

Ce rapport est divisé en quatre parties principales:

- Partie 1: Un aperçu général du projet.
- Partie 2: Une présentation sur l'état de l'art.
- Partie 3: On se concentrera sur l'identification de l'approche utilisée pour régler le problème.
- Partie 4: Une conclusion.

1.5 Cahier de charge :

1.5.1 Aperçu du projet

1.5.1.1 Resume du projet

L'analyse de sentiment est une tâche de traitement automatique des langues et d'extraction d'information. Pour un texte donné, la polarité du texte doit être identifiée en tant que positive ou négative. Dans notre projet, nous considérons les textes comme faisant partie de deux classes seulement: soit le texte est positif, soit celui-ci est négatif. L'analyse des sentiments va aider de nombreuses organisations à spécifier ce que leurs clients préfèrent et/ou détestent par conséquent ils vont améliorer leurs produits pour une bonne expérience d'utilisateur.

1.5.1.2 Buts et objectifs du projet et résultats opérationnels

L'objectif de ce projet est d'analyser les sentiments des tweets arabes en utilisant les techniques du Deep Learning. En outre, le but sera de parvenir à classifier à l'aide

de différents modèles disponibles en Python, les sentiments des tweets arabes selon qu'ils soient plutôt positifs ou négatifs, et enfin réaliser une interface graphique.

| N° | Buts | Objectifs | Résultats opérationnels |
|----|--|--|---|
| 1 | Analyse des sentiments des tweets arabe. | Utilisation des techniques de Deep Learning avec le langage Python. | Algorithme qui permet de spécifier les sentiments des tweets. |
| 2 | Réalisation d'une interface graphique | Utilisation des librairies de python pour réaliser une interface graphique | Interface graphique facilitant l'analyse des tweets |

Table 1.1: Buts et objectifs du projet

1.5.1.3 Risques et contraintes du projet

- **Contrainte de temps :** Temps très limité pour la réalisation de projet.
- **Contraintes de qualité :** Un trop grand nombre de modifications apportées au projet.

1.5.1.4 Produits livrables

Réalisation d'une application qui a la capacité de classifier les tweets arabes en deux catégories: positive ou négative.

1.5.1.5 Jalons et planning prévisionnels

Dans la présente section, on indique les aspects ou les événements importants du projet (phases, étapes, points de décision et approbation d'un produit livrable).

1.5.2 Organisation du projet

1.5.2.1 Structure de l'équipe de projet

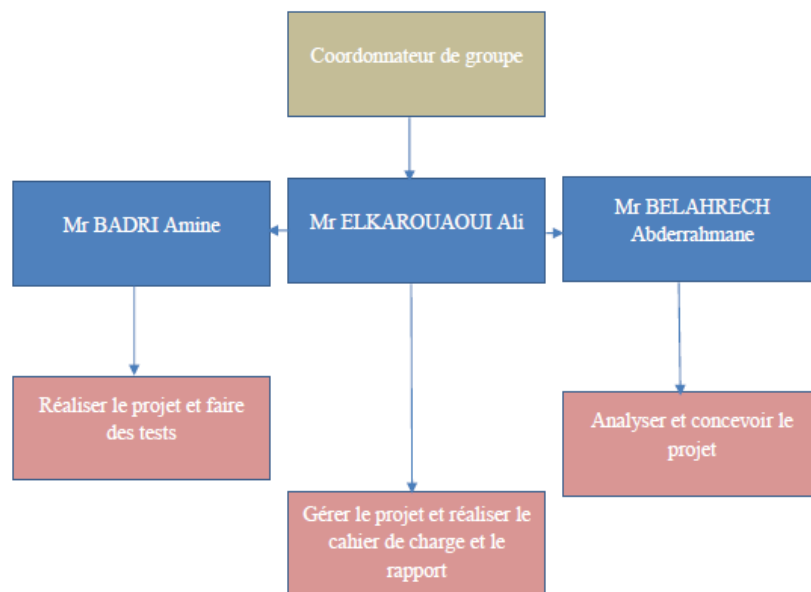


Figure 1.2: Organnigramme presentant l'équipe

1.5.2.2 Rôles et responsabilités

Dans cette section on va définir les rôles et responsabilités qui ont été confiées à chaque membre d'équipe de projet, aux intervenants et identifier les groupes de travail qui influencent considérablement le projet.

1.5.3 Spécifications fonctionnelles

Les spécifications fonctionnelles de cette application sont :

- Traitement des tweets.
- Donner des prédictions.

| Rôle dans le cadre du projet | Responsabilités | Assigné à |
|------------------------------|--|--|
| Gestionnaire de projet | Il mène les projets à bon port et coordonne tous les acteurs concernés par le projet | Encadrant et Tuteur |
| Analyste du projet | Responsable de la gestion et de la conception de projets | Mr. elkarouaoui, mr belahrech et mr badri |
| Comité d'examen du projet | L'évaluation de projet de recherche | Mr elkarouaoui, mr belahrech, mr badri et tuteur |

Table 1.3: Rôles et responsabilités

1.5.4 Spécifications techniques

Dans cette section on a écrit tous les moyens technologiques que nous avons utilisé pour réaliser notre projet :

| Solution | Critère |
|-----------|--|
| Python | Langage le plus utilisé en science de données |
| Numpy | Bibliothèque python permettant la manipulation des matrices... |
| Pandas | Bibliothèque python permettant la manipulation et l'analyse des données |
| Sklearn | Bibliothèque libre Python destinée à l'apprentissage automatique |
| Colab | C'est un environnement particulièrement adapté au machine learning, à l'analyse de données et à l'éducation. |
| Streamlit | Bibliothèque pour créer des applications web |

Table 1.4: Spécifications techniques

2

Partie théorique

Sommaire

| | |
|--|-----------|
| 2.1 Intelligence Artificielle | 22 |
| 2.2 Machine learning | 22 |
| 2.2.1 Type d'apprentissage automatique: | 22 |
| 2.2.2 Algorithmes d'apprentissage automatique: | 23 |
| 2.3 Deep learning | 24 |
| 2.3.1 Réseaux neurones | 24 |
| 2.3.2 Le modèle du perceptron | 25 |
| 2.3.3 MLP | 26 |
| 2.3.4 Les réseaux de neurones profonds | 27 |
| 2.3.4.1 L'intérêt des architectures profondes | 27 |
| 2.3.4.2 Réseaux de neurones convolutifs | 28 |
| 2.3.4.3 Réseaux de neurones récurrents | 28 |
| 2.4 L'analyse des sentiments | 29 |
| 2.4.1 Définition | 29 |

| | | |
|------------|--|-----------|
| 2.4.2 | Fonctionnement de l'analyse des sentiments | 30 |
| 2.4.3 | Les avantages de l'analyse des sentiments | 30 |
| 2.4.4 | L'Analyse des Sentiments sur les Réseaux Sociaux | 31 |
| 2.4.5 | L'Analyse des Sentiments avec Twitter | 32 |
| 2.5 | NLP | 32 |
| 2.5.1 | Nettoyage des données | 33 |
| 2.5.2 | Normalisation | 33 |
| 2.5.3 | Stemming | 34 |
| 2.5.4 | Lemmatisation | 35 |
| 2.5.5 | Tokenisation | 35 |
| 2.5.6 | Vectorisation | 35 |
| 2.6 | Conclusion | 36 |

2.1 Intelligence Artificielle

L'intelligence artificielle met en œuvre un certain nombre de techniques visant à permettre aux machines d'imiter une forme d'intelligence réelle. Exemples d'applications de l'intelligence artificielle :

- Systeme de recommandation
- Traitement d'image, son, texte...
- Voiture autonome
- Reconnaissance faciale

2.2 Machine learning

Machine Learning, aussi appelé apprentissage automatique en français, est une forme d'intelligence artificielle permettant aux ordinateurs d'apprendre sans avoir été programmés explicitement à cet effet. Cette technique permet de développer des programmes informatiques pouvant changer en cas d'exposition à de nouvelles données.

2.2.1 Type d'apprentissage automatique:

On distingue trois types d'apprentissage automatique:

- **Apprentissage supervisé :** consiste à entrer des caractéristiques et une cible comme input pour entraîner le modèle, afin de rendre l'algorithme capable, une fois entraîné, de prédire cette cible sur de nouvelles données non annotées.
Dans l'apprentissage supervisé, il existe des problèmes de classification et de régression.
 - **Classification:** consiste à prédire à quelle classe appartient quelque chose.
 - **Regression:** consiste à prédire une valeur numérique.
- **Apprentissage non supervisé :** À la différence de l'apprentissage supervisé, l'apprentissage non supervisé est celui où l'algorithme doit opérer à partir

d'exemples non annotés. En effet, dans ce cas de figure, l'apprentissage par la machine se fait de manière entièrement indépendante. Des données sont alors renseignées à la machine sans qu'on lui fournisse des exemples de résultats. L'utilisation de l'apprentissage non supervisé peut être réunie en problèmes de clustering et d'association.

- **Clustering:** Un problème de clustering est un problème pour lequel on attend de la machine qu'elle rassemble sous forme de groupe (mise en cluster) des objets présents dans des groupes de données, et ce de la manière la plus juste et efficace possible.
- **Association:** Le système d'association permet de trier et regrouper les données qui peuvent être liées grâce à certaines caractéristiques. Le but est donc de trouver des objets liés les uns aux autres sans qu'il s'agisse néanmoins d'objets identiques.
- **Apprentissage de renforcement :** En intelligence artificielle, plus précisément en apprentissage automatique, l'apprentissage par renforcement consiste à apprendre les actions à partir d'expériences, de façon à optimiser une récompense quantitative au cours du temps.

2.2.2 Algorithmes d'apprentissage automatique:

Il existe plusieurs types d'algorithmes d'apprentissage automatique parmi eux:

- **Régression linéaire simple :** C'est lorsqu'on a une seule variable explicative x et une seule variable indépendante...
- **Régression logistique :** La régression logistique utilise principalement une fonction logistique: $y = 1 / (1 + \exp(-x))$ pour modéliser une sortie binaire. Si la probabilité est supérieure à un seuil précis (souvent 0.5) les prédictions seront classées en classe 0. Sinon, la classe 1 sera attribuée.
- **Arbre de decision :** L'arbre de decision est un algorithme non paramétrique qui ressemble à un organigramme de questions à lesquelles on réponds pour prédire

la sortie .

- **Forêts aléatoires** : Composé de plusieurs arbres de décision, travaillant de manière indépendante pour résoudre un problème spécifique. L'assemblage des sorties des arbres de décision va donner une estimation globale.

2.3 Deep learning

Sous-ensemble du **Machine Learning**, le Deep Learning imite le fonctionnement du cerveau humain par des **réseaux neuronaux artificiels**. L'objectif est de créer des modèles à partir de données afin d'éclairer la prise de décision. Cette méthode peut aussi bien faire partie d'un apprentissage supervisé, semi-supervisé ou non-supervisé. Les architectures de Deep Learning ont été appliquées à des domaines tels que:

- La vision par ordinateur
- La reconnaissance vocale
- Le traitement du langage naturel
- Le filtrage des réseaux sociaux
- La bioinformatique
- La traduction automatique

Le Deep learning concerne un nombre illimité de couches de taille limitée qu'il utilise pour extraire progressivement des caractéristiques de niveau supérieur à partir de l'entrée brute. Par exemple, dans le traitement d'images, les couches inférieures peuvent identifier les bords, tandis que les couches supérieures peuvent identifier les concepts pertinents pour un humain tel que des visages.

2.3.1 Réseaux neurones

Le fonctionnement d'un réseau de neurones artificiels est calqué sur celui des neurones du cerveau humain. Il s'agit là d'une variété de technologie Deep Learning/apprentissage profond, qui fait elle-même partie de la sous-catégorie d'intelligence

artificielle du Machine Learning (apprentissage automatique).

Pour introduire les réseaux de neurones, nous allons commencer par présenter un modèle composé d'un seul neurone, appelé modèle du perceptron. Celui-ci va nous permettre de mettre en évidence les mécanismes de base de tout réseau de neurones.

2.3.2 Le modèle du perceptron

Dans sa version la plus simple, le perceptron est un réseau de neurones composé de seulement un neurone, qui prend en entrée n données binaires. Chacune de ses entrées i est pondérée par un poids noté w_i . Le neurone peut prendre les états "1" ou "0" (respectivement actif ou non actif) en fonction de ses entrées pondérées et d'un biais. Cependant ce modèle est limité car il ne peut pas résoudre des problèmes non linéairement séparables.

Équation de sortie d'un neurone formel :

$$y = f((w, x) + b) \quad (2.1)$$

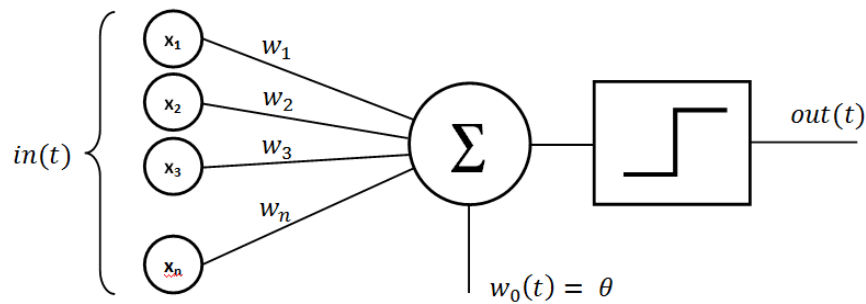


Figure 2.1: Schéma qui montre un perceptron simple

2.3.3 MLP

Les perceptrons multicouches, appelés aussi MLP (pour Multilayer Perceptron), sont des réseaux de neurones plus généraux que le perceptron. Ils sont composés d'une multitude de neurones interconnectés et organisés en couches successives. Un MLP peut être représenté par un graphe acyclique dans lequel chaque noeud représente un neurone. Les arcs orientés représentent les relations entre chaque neurone : un arc du noeud i au noeud j signifie que le neurone j prend la valeur d'activation du neurone i en entrée. La figure 2.2 montre une représentation graphique d'un MLP

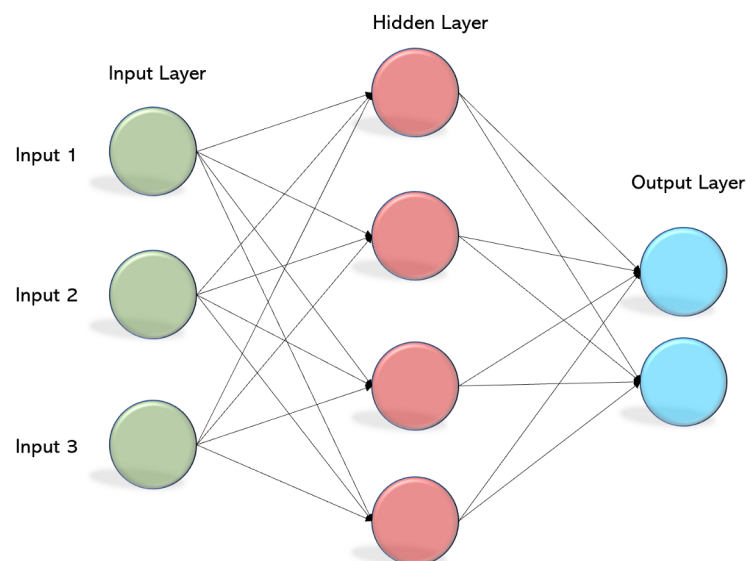


Figure 2.2: Schéma qui montre une représentation graphique d'un MLP

2.3.4 Les réseaux de neurones profonds

Les réseaux de neurones ont très longtemps été limités dans leurs architectures, en particulier concernant leur profondeur, c'est-à-dire, le nombre de couches qu'ils pouvaient apprendre.

Cette limitation s'est effondrée dans les années 2010-2012 avec l'arrivée de bases de données bien plus grandes (*Big Data*) accompagnées de capacités de calcul et de stockage plus importantes. Cette avancée a également été permises par des architectures de réseaux de neurones différentes, plus faciles à apprendre et mieux adaptées à certains types de données.

2.3.4.1 L'intérêt des architectures profondes

Dans les algorithmes d'apprentissage classiques, des caractéristiques doivent être extraites des données brutes afin d'effectuer la tâche d'apprentissage. Le but était d'avoir une représentation plus haut niveau des données. Par exemple, dans le domaine de l'analyse d'image, une première étape consiste à calculer les points d'intérêts (comme les SIFT) et les regrouper dans des sacs de mots (ou bag of word en anglais) pour entraîner un modèle classique d'apprentissage tel qu'un arbre de décision, un SVM, une forêt d'arbres aléatoires ou même un réseau de neurones. L'extraction de caractéristiques à partir des données brutes demande des bonnes connaissances sur celles-ci et sur la tâche d'apprentissage, ainsi qu'un travail d'ingénierie pour adapter les méthodes d'extraction.

Cette opération est relativement coûteuse à la mise en place, dépend du contexte et une mauvaise extraction des caractéristiques mène à de très mauvaises performances en terme d'apprentissage. L'idée des architectures profondes consiste à intégrer cette extraction de caractéristiques, normalement faite "à la main", par un processus d'apprentissage dans les premières couches du réseau de neurones. Le terme profond réfère donc au nombre de couches des réseaux de neurones profonds entre l'entrée et la sortie. Un réseau avec une seule couche cachée est appelé réseau peu profond, et un réseau avec plus de 2 couches cachées est dit profond. De nos jours, il est possible de trouver des réseaux avec une centaine, voir un millier de couches pour les plus

profonds.

2.3.4.2 Réseaux de neurones convolutifs

Les réseaux de neurones convolutifs (CNN) ont été introduits par Lecun et al. La particularité des CNN est l'utilisation de l'opération de convolution dans les premières couches intermédiaires du réseau de neurones. À l'origine, cette opération est utilisée comme filtre dans le domaine de l'image ou du son afin de mettre en évidence des motifs ou réduire un type de bruits.

Un CNN classique est généralement composé de quatre types de couches :

- Les couches convolutives, qui contiennent plusieurs opérations de convolutions appliquées sur la même entrée,
- Les couches d'opérations de mise en commun
- Les couches d'activations,
- Les couches toutes connectées.

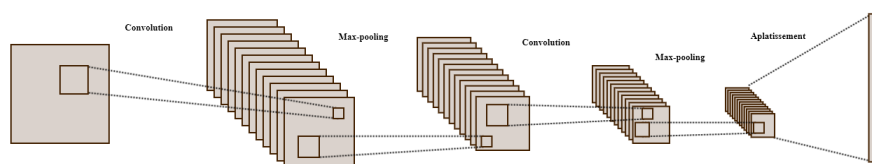


Figure 2.3: Schema qui montre une représentation graphique d'un CNN

2.3.4.3 Réseaux de neurones récurrents

Alors que les CNN sont principalement utilisés pour faire ressortir des relations spatialement proches (comme des relations entre pixels proches dans une image), les réseaux de neurones récurrents (RNN) ont été développés afin de garder un contexte temporel pour chaque événement en entrée. Ils ont été particulièrement utilisés pour de l'analyse des séries temporelles, des données audio, ou des textes dans lesquelles le

contexte est important afin d'analyser chaque nouvelle entrée.

L'idée consiste à garder des informations au cours du temps à l'intérieur des couches de neurones afin de donner un contexte aux données analysées. La sortie du RNN, à un instant t , va dépendre non seulement de l'entrée à l'instant t mais également de l'état du RNN calculé à l'instant $t-1$.

Dans sa version la plus simple, une couche d'un RNN peut être décrite comme une couche toute connectée l qui prend en entrée la couche précédente $l-1$ à l'instant t concaténée à la sortie d'elle-même (c'est-à-dire, couche l) à l'instant $t-1$.

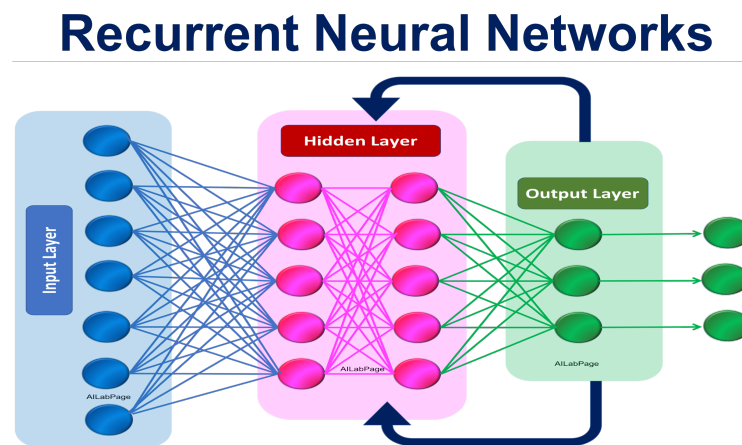


Figure 2.4: Schema qui montre une représentation graphique d'un RNN

2.4 L'analyse des sentiments

Dans cette section nous allons parler de L'Analyse des Sentiments aussi appelée Opinion Mining.

2.4.1 Définition

C'est un domaine du NLP et une interprétation et la classification des sentiments (positives; négatives et neutres) dans les données de texte à l'aide de techniques d'analyse de texte. L'analyse des sentiments permet de déterminer le sentiment qui se cache derrière une série de mots. Par exemple il permet aux entreprises d'identifier le sentiment des clients envers les produits, les marques ou les services dans les

conversations et les commentaires.

La figure suivante illustre une analyse des sentiments par des émojis:



Figure 2.5: L'analyse des sentiments (Positive, Neutre, Negative)

2.4.2 Fonctionnement de l'analyse des sentiments

L'analyse des sentiments est une technique basée sur le traitement automatique du langage humain (appelé plus communément Traitement du Langage Naturel).

L'analyse des sentiments va de la détection des émotions (colère, bonheur, peur) au sarcasme et à l'intention (p. ex. plaintes, rétroaction, opinions). Dans sa forme la plus simple, l'Analyse des Sentiments attribue ensuite une polarité (positive, négative, neutre) à un texte.

2.4.3 Les avantages de l'analyse des sentiments

On estime que 80organisées. D'énormes quantités de données textuelles (e-mails, tickets d'assistance, chats, conversations sur les réseaux sociaux, sondages, articles, documents, etc.) sont créées chaque jour, mais il est difficile d'analyser, de comprendre et de trier, sans parler de temps et d'argent. Cependant, Analyse des Sentiments aide les entreprises à comprendre tout ce texte non structuré en le marquant automatiquement.

Les avantages de l'Analyse des Sentiments comprennent:

- **Traitement des données à grande échelle:** peut-on imaginer de trier manuellement des milliers de Tweets, conversations de support client ou avis clients ? Il y a tout simplement trop de données à traiter manuellement. L'Analyse des

Sentiments aide les entreprises à traiter d'énormes quantités de données de manière efficace et rentable.

- **Analyse en temps réel:** L'Analyse des Sentiments peut identifier les problèmes critiques en temps réel, par exemple une crise de relations publiques sur les réseaux sociaux s'intensifie-t-elle ? Un client en colère est-il sur le point de se retourner ? Les modèles d'Analyse des Sentiments peuvent aider à identifier immédiatement ce type de situations, afin que d'agir immédiatement.
- **Critères cohérents:** On estime que les gens ne s'entendent que 60 à 65% le sentiment d'un texte particulier. Le marquage de texte par sentiment est très subjectif, influencé par des expériences personnelles, des pensées et des croyances. En utilisant un système centralisé d'analyse des sentiments, les entreprises peuvent appliquer les mêmes critères à toutes leurs données, ce qui les aide à améliorer la précision et à obtenir de meilleures informations.

2.4.4 L'Analyse des Sentiments sur les Réseaux Sociaux

L'Analyse des Sentiments a reçu beaucoup d'attention non seulement de la part de la recherche scientifique mais aussi par les domaines de la publicité et du marketing. Le développement du web 2.0 a entraîné un intérêt de ces domaines pour les équipes marketing, souvent soumises à un déluge de données. La solution se connecte d'elle-même aux différents réseaux sociaux (Facebook, Twitter, LinkedIn, etc.), blogs, forums, commentaires d'articles, etc. Au fur et à mesure de l'indexation des données trouvées, la solution d'Analyse des Sentiments détermine par un système de notation si le contenu global recueilli est positif, négatif ou neutre. Elle peut même être capable d'identifier des propos sarcastiques ou ironiques.

Aussi la rapidité du relais de l'information, les grandes masses de données réelles issues des réseaux sociaux sont largement utilisées pour l'analyse des sentiments. Analyser les messages récents issus des réseaux sociaux pourrait donner l'opinion générale des utilisateurs envers un sujet spécifique .

Voici comment Nick Martin, expert mondial de l'engagement sur les médias sociaux

chez Hootsuite, définit le sentiment sur les médias sociaux : « Le sentiment sur les médias sociaux correspond à l'impression positive ou négative qui se dégage d'une publication ou d'une interaction»

2.4.5 L'Analyse des Sentiments avec Twitter

Twitter est une plate-forme de communication basée sur le Web, qui permet à ses abonnés de diffuser des messages appelés « tweets » de 280 caractères maximum, leur permettant de partager des pensées, des liens ou des images. Par conséquent, Twitter est une source riche de données pour l'exploration d'opinion et l'analyse de sentiment. La simplicité d'utilisation et les services offerts par la plate-forme Twitter lui permettent d'être largement utilisée dans le monde arabe . Cette popularité nous donne accès à une mine riche d'informations qui peuvent servir comme base de données à l'analyse des tweets, qui nous fournissent des informations précieuses.



Figure 2.6: Logo de Twitter

2.5 NLP

Natural language processing est un sous-domaine de la linguistique, de l'informatique et de l'intelligence artificielle qui s'intéresse aux interactions entre les ordinateurs et le langage humain, en particulier la façon de programmer les ordinateurs pour traiter et analyser de grandes quantités de données en langage naturel.

| Valeur pour remplacer | Valeur Remplacer par |
|-----------------------|----------------------|
| ! | ا |
| ا | ا |
| آ | ا |
| ة | ة |
| " " | " " |
| "_" | " " |
| "/" | " " |
| "+" | " " |
| "=" | " " |
| "x" | " " |
| "." | " " |
| "،" | " " |
| و | و |
| يا | يا |
| " " | " " |
| "_" | " " |
| " " | " " |
| "ى" | ي |
| "\\" | " " |
| "n" | " " |
| "t" | " " |
| '؟' | '؟' |
| '؟' | '؟' |
| '!' | '!' |
| "،" | " " |
| "." | " " |
| 'و' | 'و' |
| 'ي' | 'ي' |
| ' ' | ' ' |

Figure 2.7: Norrmalisation des tweets

2.5.3 Stemming

Stemming est une technique qui aide à réduire la dimensionnalité élevée de l'espace des fonctionnalités dans la classification de textes. Plusieurs approches de Stemming existent pour la langue arabe, chacune produisant un ensemble différent de racines. En arabe, les approches de Stemming les plus connues utilisées sont root-based Stemmers, Arabic Light Stemming

pour cette étape nous avons utilisé Arabic Light Stemming. Cette approche n'est pas utilisée pour produire la racine linguistique d'une forme de surface arabe donnée, mais pour supprimer les suffixes et préfixes les plus fréquents. Les suffixes le plus courant comprend les duals et les pluriels pour les formes masculines et féminines,

possessives, les articles définis et les pronoms.

L'exemple suivant illustre ce principe : Prenons cette liste des mots ou Tokens ['دعاء', 'إلى', 'قول', 'أن', 'ود', 'م', 'ال', 'نحو', 'الذين', 'نحن'] après l'utilisation du Light Stemming elle devient ['دعاء', 'إلى', 'قول', 'أن', 'ود', 'م', 'ال', 'نحو', 'الذين', 'نحن']

2.5.4 Lemmatisation

Lemmatisation : La tâche de supprimer uniquement les terminaisons flexionnelles et de renvoyer la forme de dictionnaire de base d'un mot également connu sous le nom de lemme.

== exemple:

2.5.5 Tokenisation

La tokenisation cherche à transformer un texte en une série de tokens individuels. Dans l'idée, chaque token représente un mot, et identifier des mots semble être une tâche relativement simple. Exemple:

La phrase:

Premier mot:

2.5.6 Vectorisation

Dans le traitement automatique du langage naturel, nous voulons créer des programmes informatiques qui fonctionnent avec les langages humains, mais les machines ne comprennent que les nombres!

Pour cette raison, nous devons associer ces mots (parfois même les phrases) à des vecteurs, c'est ce qu'on appelle la vectorisation de texte.

Il existe plusieurs techniques de vectorisation. Parmi eux :

- **Sac de mots** : Un sac de mots est une représentation de texte qui décrit l'occurrence de mots dans un document. Nous gardons simplement une trace du nombre de mots et ne tenons pas compte des détails grammaticaux et de l'ordre des mots.

- **Vectorisation TF-IDF** : TF-IDF signifie Term Frequency — Inverse Document Frequency et est une statistique qui vise à mieux définir l'importance d'un mot pour un document, tout en tenant compte de la relation avec d'autres documents du même corpus.

2.6 Conclusion

Dans ce chapitre nous avons montré un aperçu de l'état de l'art de l'apprentissage profond. Celui-ci est particulièrement riche et le nombre d'applications augmente constamment depuis quelques années. Les réseaux de neurones peuvent être appris de manière supervisée ou non-supervisée. L'apprentissage se fait par une descente de gradient (ou une variante de cette méthode) sur une base de données la plus large possible. Nous allons voir dans le chapitre suivant la phase de réalisation de projet.

3

Realisation

Sommaire

| | |
|--|-----------|
| 3.1 Objectives | 39 |
| 3.2 Outils | 39 |
| 3.2.1 Language de programmation utilisé (Python) | 39 |
| 3.2.2 L'IDE utilisé (Pycharm) | 40 |
| 3.2.3 Google Colab | 41 |
| 3.2.4 Scikit-learn | 41 |
| 3.2.5 Pandas | 42 |
| 3.2.6 Numpy | 42 |
| 3.2.7 Tensorflow | 43 |
| 3.2.8 Keras | 43 |
| 3.2.9 Matplotlib | 44 |
| 3.2.10 Re | 44 |
| 3.2.11 NLTK | 44 |
| 3.2.12 Streamlit | 44 |

| | |
|--|-----------|
| 3.3 Application | 45 |
| 3.3.1 Dataset | 45 |
| 3.3.1.1 Echantillon du dataset | 45 |
| 3.3.1.2 Nombre des tweets negatives et positives | 46 |
| 3.3.2 Pretraitement des donnees | 46 |
| 3.3.3 Resultat des models | 48 |
| 3.3.4 Application web | 49 |
| 3.4 Conclusion | 50 |

3.1 Objectives

L'objectif generale de ce projet est de realiser une application permettant de classifier les tweets arabes en deux classes (Positif ou negatif). Voici les objectifs principales de cette application:

- Trouver une dataset contenant des tweets arabes annotées.
- Comparer les differents model afin de choisir le meilleur.
- Realiser une application web facilitant la tache de classification.

3.2 Outils

3.2.1 Language de programmation utilisé (Python)

Phyton est un langage de programmation interprété, multi-paradigme et multi-plateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions ; il est ainsi similaire à Per, Ruby, Scheme, Smalltalk et Tcl.

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

Il est également apprécié par certains pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation.



Figure 3.1: Logo de python

3.2.2 L'IDE utilisé (Pycharm)

Nous avons utilisé pour notre programmation l'IDE Pycharm pour implémenter notre projet en langage Python. PyCharm est un environnement de développement intégré utilisé pour programmer en Python.

- Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciel de gestion de versions, et supporte le développement web avec Django.
- Développé par l'entreprise tchèque JetBrains, c'est un logiciel multi-plateforme qui fonctionne sous Windows, Mac OS X et Linux. Il est décliné en édition professionnelle, diffusé sous licence propriétaire, et en édition communautaire diffusé sous licence Apache.

-

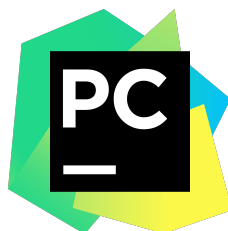


Figure 3.2: Logo de pycharm

3.2.3 Google Colab

Colab (ou "Colaboratory") vous permet d'écrire et d'exécuter du code Python dans votre navigateur avec aucune configuration requise accès gratuit aux GPU partage facile que vous soyez étudiant, data scientist ou chercheur en IA. Colab permet à n'importe qui d'écrire et d'exécuter le code Python de son choix par le biais du navigateur. C'est un environnement particulièrement adapté au machine learning, à l'analyse de données et à l'éducation.



Figure 3.3: Logo de colab

3.2.4 Scikit-learn

elle dispose d'un large éventail d'algorithmes d'apprentissage supervisés et non supervisés qui fonctionnent sur une interface cohérente en Python. La bibliothèque peut également être utilisée pour l'exploration de données et l'analyse de données. Les principales fonctions d'apprentissage automatique que la bibliothèque Scikit-learn peut gérer sont la classification, la régression, la mise en cluster, la réduction de dimensionnalité, la sélection de modèle et le prétraitement.



Figure 3.4: Logo de Sklearn

3.2.5 Pandas

elle est en train de devenir la bibliothèque Python la plus populaire utilisée pour l'analyse de données avec prise en charge de structures de données rapides, flexibles et expressives conçues pour fonctionner à la fois sur des données «relationnelles» ou «étiquetées».



Figure 3.5: Logo de Pandas

3.2.6 Numpy

elle est une extension du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

Plus précisément, cette bibliothèque logicielle libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynômes.



Figure 3.6: Logo de Numpy

3.2.7 Tensorflow

TensorFlow est un outil open source d'apprentissage automatique développé par Google. Le code source a été ouvert le 9 novembre 2015 par Google et publié sous licence Apache. Il est fondé sur l'infrastructure DistBelief, initiée par Google en 2011, et est doté d'une interface pour Python, Julia et R. TensorFlow est une plate-forme Open Source de bout en bout dédiée au machine learning. Elle propose un écosystème complet et flexible d'outils, de bibliothèques et de ressources communautaires permettant aux chercheurs d'avancer dans le domaine du machine learning, et aux développeurs de créer et de déployer facilement des applications qui exploitent cette technologie.



Figure 3.7: Logo de Tensorflow

3.2.8 Keras

Keras est une API conçue pour les êtres humains, pas pour les machines. Keras suit les meilleures pratiques pour réduire la charge cognitive : il propose des API cohérentes et simples, il minimise le nombre d'actions de l'utilisateur requises pour les cas d'utilisation courants et il fournit des messages d'erreur clairs et exploitables. Il contient également une documentation complète et des guides de développement.



Figure 3.8: Logo de Keras

3.2.9 Matplotlib

Matplotlib est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques. Elle peut être combinée avec les bibliothèques python de calcul scientifique NumPy et SciPy.



Figure 3.9: Logo de Matplotlib

3.2.10 Re

Une expression régulière (ou RE) spécifie un ensemble de chaînes qui lui correspond ; les fonctions de ce module vous permettent de vérifier si une chaîne particulière correspond à une expression régulière donnée (ou si une expression régulière donnée correspond à une chaîne particulière, ce qui revient au même). Ce module fournit des opérations de mise en correspondance d'expressions régulières similaires à celles trouvées dans Perl.

3.2.11 NLTK

NLTK est une plate-forme leader pour la création de programmes Python pour travailler avec des données de langage humain. Il fournit des interfaces faciles à utiliser à plus de 50 corpus et ressources lexicales telles que WordNet, ainsi qu'une suite de bibliothèques de traitement de texte pour la classification, la tokenisation, la radicalisation, le balisage, l'analyse et le raisonnement sémantique, des wrappers pour les bibliothèques NLP de puissance industrielle, et un forum de discussion actif.

3.2.12 Streamlit

Streamlit est un framework open-source Python spécialement conçu pour les ingénieurs en machine learning et les Data scientists. Ce framework permet de créer des

applications web qui pourront intégrer aisément des modèles de machine learning et des outils de visualisation de données. Contrairement aux autres framework python (Dash, ...) pour créer des applications autour de la données, Streamlit permet de créer de belles applications web sans écrire du code HTML. Ce framework permet aussi d'avoir des applications performantes grâce à la mise en cache via une annotation.



Figure 3.10: Logo de Streamlit

3.3 Application

3.3.1 Dataset

On a utilisé une dataset contenant 47000 tweets arabes contenant deux classes **negatives** et **positives**.

3.3.1.1 Echantillon du dataset

Voici les cinq premières cellules du dataset:

| | label | tweet |
|---|-------|--|
| 0 | pos | ...نحن الذين يتحول كل ما نود أن نقوله إلى دعاء للـ |
| 1 | pos | ...وفي النهاية لن يبقى معك أحد إلا من رأى الجمال |
| 2 | pos | من الخير نفسه 🍷 |
| 3 | pos | ...زلزل الملعب_نصرنا_يلعب كن عالي الهمة ولا ترض # |
| 4 | pos | ...الشيء الوحيد الذي وصلوا فيه للعالمية هو : المس |

Figure 3.11: Cinq premiers cellules du dataset

3.3.1.2 Nombre des tweets negatives et positives

```

fig, axes = plt.subplots(1, 2, figsize=(15, 7), sharey=False)
fig.suptitle('Data Exploration - Dataset Balance')

//----- COUNT PLOT -----

axes[0].set_title("Count Chart")
sns.countplot(ax=axes[0], data=training_data, hue='label', x='label')
axes[0].legend(title = "neg vs pos:", loc='upper right')

//----- PIE CHART -----
----

axes[1].set_title("Pie Chart")
data_count = [training_data[training_data["label"]=="neg"].count().label,
               training_data[training_data["label"]=="pos"].count().label]
keys = ['Negative', 'Positive']

palette_color = sns.color_palette('deep')
axes[1].pie(data_count, labels=keys, colors=palette_color,
            autopct='%0f%%')
axes[1].legend(title = "neg vs pos:", loc='upper right')

```

Figure 3.12: Code pour realiser les plots suivants

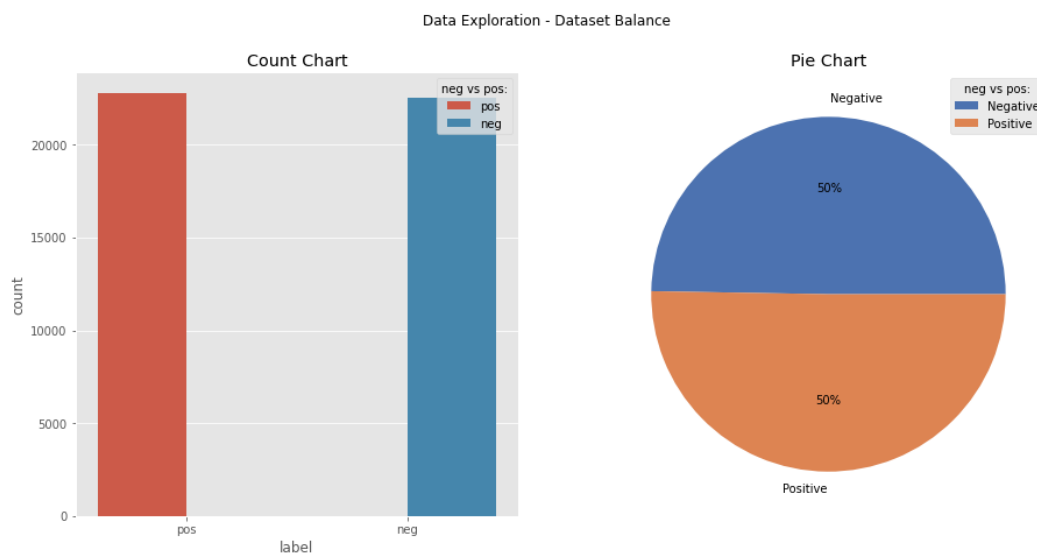


Figure 3.13: Nombre des tweets negatives et positives

3.3.2 Pretraitement des donnees

On avait utilisé plusieurs bibliothèques afin de nettoyer et pretraiter les données à savoir **Nltk**, **Re**, **Pyarabic**, **Sklearn**. Voici un echantillon du dataset avans et apres le

pretraitement:

| | label | tweet |
|-------|-------|--|
| 0 | pos | ...نحن الذين يتحول كل ما نود أن نقوله إلى دعاء للـ |
| 1 | pos | ...وفي النهاية لن يبقى معك أحداً من رأي الجمال |
| 2 | pos | 💛 من الخير نفسه |
| 3 | pos | ...زلزل_الملعب_نصرنا_يلعب كن عالي الهمة ولا ترض# |
| 4 | pos | ...الشيء الوحيد الذي وصلوا فيه للعالمية هو : المس |
| ... | ... | ... |
| 45261 | neg | 👉 مش هتلاقي |
| 45262 | neg | ❤ استغفر الله العظيم والله عصة |
| 45263 | neg | ...انا مستحيل بقدر وقف بوظة وشوكولا وشاورما هول ا |
| 45267 | neg | ❤ اكره مره اني اكون حارس وحظي دايم يخليني كذا |
| 45271 | neg | ❤ احسبك على الاليم |

Figure 3.14: Dataset avant pretraitement

| | label | tweet |
|-------|-------|---|
| 0 | pos | ...يتحول نود ان نقوله الي دعاء الله تبحثوا فينا فو |
| 1 | pos | ... وفي النهاية يبقى معك احداً رأي الجمال روحك |
| 2 | pos | الخير نفسه |
| 3 | pos | ...زلزل_الملعب_نصرنا_يلعب عالي الهمة ترضي بخير القم |
| 4 | pos | ... الشيء الوحيد وصلوا للعالمية الميسار تري كانوا |
| ... | ... | ... |
| 45261 | neg | مش هتلاقي |
| 45262 | neg | استغفر الله العظيم والله عصه |
| 45263 | neg | ...انا مستحيل بقدر وقف بوظه وشوكولا وشاورما هول ا |
| 45267 | neg | اكره مره اني اكون حارس وحظي دايم يخليني |
| 45271 | neg | احسبك على الاليم |

Figure 3.15: Dataset apres pretraitement

3.3.3 Resultat des models

Apres l'évaluation des models (SVC, Logistic Regression, Random Forest, MLP, Naive Bayes, CNN) voila le resultat:

| MLP | | | | |
|----------------------------------|-----------|--------|----------|---------|
| Accuracy score is 0.93 | | | | |
| score with Random forest : | | | | |
| | precision | recall | f1-score | support |
| neg | 0.96 | 0.73 | 0.83 | 4455 |
| pos | 0.92 | 0.99 | 0.96 | 14459 |
| accuracy | | | 0.93 | 18914 |
| macro avg | 0.94 | 0.86 | 0.89 | 18914 |
| weighted avg | 0.93 | 0.93 | 0.93 | 18914 |
| Logistic Regression | | | | |
| Accuracy score is 0.80 | | | | |
| score with logistic regression : | | | | |
| | precision | recall | f1-score | support |
| neg | 0.56 | 0.77 | 0.65 | 4455 |
| pos | 0.92 | 0.81 | 0.86 | 14459 |
| accuracy | | | 0.80 | 18914 |
| macro avg | 0.74 | 0.79 | 0.75 | 18914 |
| weighted avg | 0.83 | 0.80 | 0.81 | 18914 |

Figure 3.16: Resultat du MLP et Logistic regression

| RandomForest | | | | |
|----------------------------|-----------|--------|----------|---------|
| Accuracy score is 0.95 | | | | |
| score with Random forest : | | | | |
| | precision | recall | f1-score | support |
| neg | 0.96 | 0.81 | 0.88 | 4455 |
| pos | 0.94 | 0.99 | 0.97 | 14459 |
| accuracy | | | 0.95 | 18914 |
| macro avg | 0.95 | 0.90 | 0.92 | 18914 |
| weighted avg | 0.95 | 0.95 | 0.95 | 18914 |
| NaiveBayes | | | | |
| Accuracy score is 0.83 | | | | |
| score with Naive Bayes : | | | | |
| | precision | recall | f1-score | support |
| neg | 0.61 | 0.76 | 0.68 | 4455 |
| pos | 0.92 | 0.85 | 0.88 | 14459 |
| accuracy | | | 0.83 | 18914 |
| macro avg | 0.76 | 0.80 | 0.78 | 18914 |
| weighted avg | 0.85 | 0.83 | 0.83 | 18914 |

Figure 3.17: Resultat du Random Forest et Naive Bayes


```
----- SVC -----
Accuracy score is 0.82
score with Naive Bayes :
      precision    recall  f1-score   support

     neg         0.58         0.80         0.67         4455
     pos         0.93         0.82         0.87        14459

 accuracy                   0.82        18914
 macro avg              0.76         0.81         0.77        18914
 weighted avg           0.85         0.82         0.83        18914
```

Figure 3.18: Resultat du SVC

```
208/208 [=====] - 224s 1s/step - loss: 0.3864 - accuracy: 0.7954 - val_loss: 0.2097 - val_accuracy: 0.9026
Epoch 2/2
208/208 [=====] - 222s 1s/step - loss: 0.1067 - accuracy: 0.9637 - val_loss: 0.2431 - val_accuracy: 0.8968
277/277 [=====] - 14s 51ms/step - loss: 0.1907 - accuracy: 0.9249
Model: "sequential"
```

Figure 3.19: Resultat du CNN

3.3.4 Application web

On a créer une application web permettant de classifier les tweets arabes en utilisant **Streamlit** comme **framework** et **Ngrok** pour acceder à l'application et visualiser le resultat depuis colab.

SENTIMENT ANALYZER

Text to analyze

اكره مره اني اكون حارس وحظي دايم يخيليني كذا ❤️

Classify

Negative

Figure 3.20: L'application web

3.4 Conclusion

A travers ce chapitre, nous avons présenté la réalisation de l'application en représentant quelques interfaces graphiques que nous avons jugé les plus importantes et en décrivant.

4

Conclusion

Dans ce projet, nous avons appris beaucoup de choses que nous n'avions jamais étudiées auparavant, comme AI, ML, DL, ... etc. Nous avons utilisé ces compétences dans un sujet sensible, qui est la classification des tweets arabes. Au cours de ce projet final, nous avons rencontré de nombreux problèmes, comme trouver le meilleur algorithme pour les tweets et même lorsque nous avons trouvé le bon, nous avons rencontré des problèmes pour former et tester le modèle, en raison de notre manque d'expérience dans l'apprentissage en profondeur. Mais heureusement, nous avons atteint nos objectifs et nous avons développé notre application pour classer les sentiments des tweets arabes.

Bibliographie

- <https://colab.research.google.com/>
- <https://www.openclassrooms.com>
- Dataset link.
- <https://www.sololearn.com/learning/1094>
- <https://www.youtube.com/c/Freecodecamp>
- <https://docs.streamlit.io/>
- <https://stackoverflow.com/>