République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique

UNIVERSITE SAAD DAHLEB BLIDA 1
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

Mémoire de Fin d'Etudes Pour l'obtention Du Diplôme de Master en Informatique

Option : Ingénierie Des Logiciels

THEME

La proposition d'un nouveau modèle pour calculer le degré de la dépression à partir des réseaux sociaux

Présenté par :

Ghribi Asma Limani Fatma

Soutenu devant le jury

Présidente :	Mme.Oukid Lamia
Examinatrice :	Mme.Ferdi Imene
Promotrice :	Mme.Boumahdi Fatima
Co-promotrice :	Mme.Madani Amina

Année scolaire : 2020-2021

Remerciement

Tout d'abord nous remercions Allah qui nous a donné la force et la patience d'accomplir ce travail dans les meilleures conditions.

Mes plus sincères remerciements, à notre promotrice Dr.Boumahdi Fatima pour son attention, son orientation, son aide pendant la réalisation de ce travail et pour être source d'information et de communication sans hésiter à aucun moment de consacrer une part de son temps précieux .Nous remercions Dr.Madani Amina pour son aide et ses conseils pour élaborer ce travail.

Nous souhaitons adresser nos remerciements les plus sincères à M. Hentabli Hamza, qui nous a aidé et nous a beaucoup dirigé coté programmation.

Nos sincères remerciements à tous les membres jury qui nous ont fait l'honneur de réviser ce travail.

Dédicace

Je dédie ce travail,

***A mes très chers parents**, pour leurs efforts et leurs sacrifices durant toute ma vie, leurs encouragement et soutiens pour persévérer pendant mes études jusqu'à l'aboutissement de ce travail.*

***A mes très chères sœurs Asmaa, Nassima et Sarah,** qui ont été pour moi une source de joie, de courage et de bonheur.*

***A toute la famille LIMANI** et **NAAMANE**: mes tantes, mes oncles, mes cousins et cousines qui m'ont toujours souhaité la bonne chance.*

***A Ghribi Asma,** chère amie avant d'être binôme qui ont passé de belles journées ensemble à l'université.*

***A ma très chère amie Belkhir Marwa,** qui était toujours à mes côtés.*

A mes chers amis, (Houria, Khawla, Romaissa, Nadjib, Ibrahim, Mohamed, Asma…)

En souvenir de nos éclats de rire et des bons moments, en souvenir de tout ce qu'on a vécu ensemble, j'espère de tout mon cœur que notre amitié durera éternellement.

A tous mes amis de promotion de Master 2 en ingénierie des logiciels.

Fatma

Dédicace

Je dédie ce modeste travail à ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A la femme qui a souffert sans me laisser souffrir, et qui m'a simplement donné la vie. Qui n'a pas rejeté mes demandes. A celle qui est resté éveillé toute les nuits pour que je puisse dormir en toute sécurité. A la plus belle femme de ma vie. A ***ma mère****, mon amour, la source de mon bonheur, qui, si je lui donnais tout dans le monde, je ne compenserai pas sa fatigue.*

A l'homme qui m'a fourni tout ce dont j'avais besoin, à mon support et à mon héros dans ma vie. A celui qui est resté debout et a travaillé dur pour mon confort. A qui a toujours été ma source de force pour affronter divers obstacles, à ***mon père*** *bien-aimé, la créature la plus chère au monde.*

A l'homme le plus cher à mon cœur : mon ***grand-père****. Je prie Dieu Tout-Puissant de le préserver et de le garder avec nous.*

A celles qui n'ont cessé de me conseiller, de m'encourager et de me soutenir tout au long de mes études : mes chères sœurs ***Keltoum*** *et* ***Nada****. Que Dieu les protège et leurs offre la chance et le bonheur.*

A ceux qui m'ont soutenu et ils se tenaient de mon côté dans mes moments difficiles: mes frères ***Amine*** *et* ***Samir*** *et bien sur mon frère* ***Mohamed****. Que Dieu leur donné une longue et heureuse vie.*

A ceux qui sont source de bonheur et de joie dans notre famille : mes princes ***Houssem-El-Bachir*** *et* ***Tarek*** *ainsi qu'à mes princesses* ***Halima****,* ***Anfal et Soudjoud****. Puisse Dieu leur donne la santé, le bonheur et surtout la réussite.*

A qui j'appartiens pour eux et ils appartiennent à mon cœur : à toute ***ma famille*** *et* ***mes proches*** *et à tous ceux qui me donnent de l'amour et de la véracité.*

Au plus beau cadeau que m'a fait l'université, A celle avec qui j'ai partagé des moments difficiles, ainsi que des moments magnifiques que je n'oublierai jamais peu importe combien de temps je vis. A ma chère sœur de cœur et mon binôme dans cette mémoire : ***Limani Fatma****. Et à toute* ***sa famille****, qui a toujours fait preuve d'esprit de collaboration et de serviabilité, pour que notre amitié dure.*

A ***mes chers amis****, et a toute* ***la promotion Master 2 ingénieur de logiciel****, ainsi qu'à tous* ***les professeurs*** *et* ***les enseignants*** *que j'ai eu durant tout mon cursus scolaire et qui m'ont permis de réussir dans mes études. A toute personne ayant contribué à ce travail de près ou de loin.*

Asma

Résumé

Au cours des dernières années, un grand nombre de personnes ont été attirées par les plateformes de médias sociaux comme Facebook, Twitter et Reddit. La plupart d'entre eux utilisent ces moyens pour exprimer leurs sentiments, leurs croyances et leurs opinions. Par conséquent, notre travail vise à utiliser le potentiel de plateforme Reddit pour résoudre l'un des plus grands problèmes de santé mentale, c'est le calcul de degré de la dépression, sous-domaine de la détection précoce de dépression grâce à l'analyse des sentiments. Notre étude est basée sur une tentative de répondre à un questionnaire du l'inventaire de dépression de beck (BDI) de 21 questions pour chaque utilisateur. Nous avons développé deux modèles différents (CNN et la combinaison de l'Auto-encoder et CNN) dérivés de l'apprentissage en profondeur en utilisant une technique statistique pour calculer le degré de la dépression. Nos résultats finaux ont été évalués à la fin de ce mémoire et comparés aux travaux précédents.

Mots clé :

Reddit, calcul de degré de la dépression, la détection précoce, analyse des sentiments, questionnaire du l'inventaire de dépression de beck (BDI), apprentissage en profondeur, CNN, Auto-encoder.

Abstract

Over the past few years, a large number of people have been attracted to social media platforms like Facebook, Twitter, and Reddit. Most of them use these means to express their feelings, beliefs and opinions. Therefore, our work aims to use the potential of the Reddit platform to solve one of the biggest mental health issues, which is calculating the degree of depression a subfield of early detection of depression through sentiment analysis. Our study is based on an attempt to answer a Beck Depression Inventory (BDI) questionnaire of 21 questions for each user. We have developed two different models (CNN and the combination of Auto-encoder and CNN) derived from deep learning using a statistical technique to calculate the degree of depression. Our final results were evaluated at the end of this thesis and compared to the previous work.

Keywords:

Reddit, calculating the degree of depression, early detection of depression, sentiment analysis, Beck Depression Inventory (BDI) questionnaire, deep learning, CNN, Auto-encoder.

ملخص

على مدى السنوات القليلة الماضية، انجذب عدد كبير من الأشخاص إلى منصات التواصل الاجتماعي مثل فايسبوك، تويتر و ريديت. يستخدم معظمهم هذه الوسائل للتعبير عن مشاعرهم ومعتقداتهم وآرائهم. لذلك يهدف عملنا إلى استخدام إمكانات منصة ريديت لحل واحدة من أكبر مشكلات الصحة العقلية، وهي حساب درجة الاكتئاب وهو مجال فرعي من الكشف المبكر عن الاكتئاب من خلال تحليل المشاعر. تستند دراستنا إلى محاولة الإجابة على استبيان BDIالمكون من 21 سؤال لكل مستخدم. لقد طورنا نموذجين مختلفين (CNN والجمع بين Auto-encoder وCNN) مستمدين من التعلم العميق باستخدام تقنية إحصائية لحساب درجة الاكتئاب. تم تقييم نتائجنا النهائية في نهاية هذه الأطروحة ومقارنتها بالأعمال السابقة.

الكلمات المفتاحية:

ريديت، حساب درجة الاكتئاب، الكشف المبكر عن الاكتئاب، تحليل المشاعر، استبيان BDI، التعلم العميق، CNN، Auto-encoder.

Table des matières

Liste des figures

Liste des tableaux

Introduction Générale

L'analyse des sentiments a été traitée par Nasukawa et YI [1], qui est récemment devenue l'un des domaines de recherche en croissance et en développement liés au traitement du langage naturel et à l'apprentissage en profondeur. De nombreuses études ont montré que l'analyse du sentiment est très intéressante pour les personnes qui se concentrent sur l'opinion publique, pour de nombreuses raisons personnelles, commerciales ou politiques.

Elle s'est avérée bénéfique pour plusieurs tâches de traitement du langage naturel (TAL) telles que les systèmes de réponse et l'extraction de l'information. L'extraction d'informations a pour but d'extraire une information pertinente à un sujet particulier ou aux besoins de l'utilisateur.

De nos jours, les réseaux sociaux jouent un rôle très important dans la vie quotidienne car les gens cherchent à travers eux à diffuser leurs sentiments, opinions et pensées, à partir desquels diverses informations peuvent être extraites qui aident à prendre des décisions et à résoudre divers problèmes qui peuvent être basique et très sensible pour l'être humain.

La charge des troubles mentaux continue de croître et d'avoir une forte incidence sur la santé, ainsi que des conséquences majeures sur le plan social. Selon l'Organisation mondiale de la santé environ 450 millions [1] de personnes dans le monde souffrent actuellement de ces problèmes.

Parmi les troubles mentaux les plus courants, nous avons la dépression qui est l'une des principales causes de suicide dans le monde. La dépression est une maladie qui peut prendre plusieurs formes et toucher chacun d'entre nous (quels que soient son âge, son sexe, son niveau social…). Il est varié d'un simple sentiment de tristesse à une maladie extrêmement grave.

La dépression affecte davantage la qualité de la vie que la plupart des maladies physiques et dans certains cas, mène même au suicide ou à des tentatives de suicide. On a en outre démontré l'existence de liens réciproques entre les maladies physiques et la dépression. À l'échelle planétaire, près de 264 millions [2] de personnes de tous âges en souffrent. Cependant,

[1] https://www.who.int/whr/2001/en/whr01_en.pdf
[2] https://www.who.int/fr/news-room/fact-sheets/detail/mental-disorders

seule une petite minorité d'entre eux bénéficient d'un traitement de base, c'est pourquoi une détection précoce est nécessaire.

Notre problème s'intéresse à calculer le degré de dépression à partir des réseaux sociaux en proposant une technique automatique pour pouvoir traiter une énorme quantité de données et d'estimer des réponse précises et fiables en peu de temps. Dans le but de pouvoir accélère le traitement, pour réduire les conséquences des troubles dépressifs dus à l'effet de leur sévérité sur les performances d'une personne dans la vie quotidienne en termes d'humeur et de pensées, qui dans les cas extrêmes conduisent à suicide, et que les patients puissent vivre une vie saine et dynamique.

Pour traiter ce problème, nous exploitons les plateformes de réseaux sociaux plus particulièrement Reddit en raison de leur énorme quantité de texte à trouver et de leur disponibilité en se basant sur l'une des technologies les plus avancées d'aujourd'hui, qui est l'apprentissage en profondeur.

Pour cette raison, nous allons consacrer notre travail à répondre aux questions suivantes :

- Comment traiter les publications de réseaux sociaux, en gardant la sémantique ?
- Comment présenter ces publications, pour que notre modèle proposé les reconnaisse?
- Comment appliquer les approches de l'apprentissage en profondeur dans le calcul de degré de dépression ?
- Comment évaluer les résultats que nous avons obtenus ?

Dans ce projet, notre objectif principal est de réaliser un système pour calculer automatiquement le degré de dépression en utilisant un ensemble de données composer des publications d'utilisateurs dans le réseau social Reddit fournies par la conférence CLEF (Conference and Labs of the Evaluation Forum) [3] sur la détection précoce des risques sur internet.

Cette étude permet d'utiliser les techniques d'apprentissage en profondeur pour évaluer divers symptômes de la dépression en répondant sur 21 questions de l'inventaire de dépression de beck « BDI » pour toutes les publications d'utilisateurs.

[3] https://early.irlab.org/

Notre défi est d'utiliser deux modèles d'apprentissage en profondeur qui ont connu un grand succès dans le domaine des images et leurs résultats ont été étonnants, où nous cherchons à nous donner les mêmes résultats pour le texte. Nous avons développé le modèle de réseaux de neurones convolutifs (CNN) qui était utilisé l'année dernière dans le même problème de mesurer de la gravité des signes de dépression, afin d'améliorer et augmenter ses résultats. En plus de cela, nous avons proposé de combiné le modèle CNN avec le modèle auto-encoder, où c'était un grand défi pour nous car nous n'avons pas trouvé d'études antérieures à ce modèle dans les textes. Par la suite, pour générer différentes exécutions, nous avons utilisés une méthode statistique afin de mesurer le degré de dépression.

Afin de mettre en œuvre et d'atteindre notre objectif, nous avons organisé notre mémoire en quatre chapitres distincts.

Chapitre 1 : **Apprentissage en profondeur**

Dans le premier chapitre nous avons présenté l'intelligence artificielle, l'apprentissage automatique et ses différentes méthodes, puis nous passerons à l'apprentissage profond et à ses applications ensuite nous passerons en revue ses différentes techniques.

Chapitre 2 : **Etat de l'art : La mesure de degré de la dépression**

Dans ce chapitre, nous avons présenté une étude sur le traitement automatique du langage et l'analyse des sentiments, nous avons abordé ses différentes catégories, ses niveaux, ses types, ses approches, ses difficultés et ses domaines d'application. Nous avons examiné également une variété de concepts de base de la dépression. Ainsi nous avons exploré les travaux liés à la mesure de degré de dépression qui abordent un problème similaire au nôtre, afin d'effectuer une comparaison entre eux.

Chapitre 3 : **Conception des modèles pour calculer le degré de la dépression à partir des réseaux sociaux.**

Ce chapitre présentera une description sur la conception et l'architecture de notre modèle.

Chapitre 4 : **Expérimentations et résultats**

Dans ce dernier chapitre, nous avons expliqué l'environnement de travail de notre solution, les librairies utilisées, notre code source, et conclu en présentant les principaux résultats que nous avons obtenus et en les comparant avec d'autres travaux.

Enfin nous terminons notre mémoire par une conclusion générale et des perspectives à venir.

Chapitre 01 : Apprentissage en profondeur

1 Introduction

Ces dernières années, nous avons constaté une amélioration considérable des performances des systèmes d'intelligence artificielle dans de nombreux domaines [2], où l'IA comporte plusieurs sous-domaines, notamment l'apprentissage automatique. L'évolution de l'apprentissage automatique a conduit à des avancées et à des améliorations significatives dans la façon dont nous interagissons avec notre monde. L'une de ces avancées passionnantes est l'apprentissage en profondeur, qui est à l'origine de l'explosion actuelle d'IA. L'apprentissage en profondeur attire beaucoup l'attention, cela est dû au niveau de performance extraordinaire qu'il a atteint.

Dans ce chapitre, nous commençons par une introduction sur l'intelligence artificielle, l'apprentissage automatique et ses différentes méthodes. Ensuite, nous présentons notre approche utilisée qui est l'apprentissage en profondeur, les notions associées et ses applications. Enfin, nous discuterons des différentes catégories et des techniques d'apprentissage en profondeur en nous appuyant sur des exemples.

2 L'intelligence artificielle (Artificial intelligence) IA

L'intelligence artificielle est une branche de l'informatique grâce à laquelle nous pouvons créer des machines intelligentes qui peuvent se comporter et penser comme des humains et prendre des décisions par elles-mêmes [3].

L'IA est composée de deux mots, artificiels qui définissent quelque chose qui est fabriqué par les humains et l'intelligence qui fait référence à la capacité de penser par elle-même, ce qui fait de l'intelligence artificielle un "pouvoir de réflexion créé par l'homme" [3].

Ce domaine a été formé avec l'idée que les machines seraient un jour capables de penser comme les humains à travers leur intelligence et leur conscience, l'IA a non seulement été un tournant dans le domaine de la recherche, mais a également joué un rôle important dans la révolution des industries et du travail tels que nous les connaissons aujourd'hui [3].

Dans le but ultime de créer une conscience, l'IA passe par plusieurs étapes de planification, de raisonnement, d'analyse des données, de prédiction des résultats et d'action en conséquence [3]. L'IA implique également l'utilisation de statistiques et de probabilités et de divers autres outils mathématiques [3].

3 Apprentissage automatique (Machine Learning) ML

L'apprentissage automatique est un sous-domaine de l'IA [4]. Se concentre sur le développement de programmes informatiques qui peuvent accéder aux données et les utiliser pour apprendre par eux-mêmes [4].

Deux définitions classiques de ML [5] sont celle d'Arthur Samuel en 1956 qui a décrit l'apprentissage automatique comme « la capacité pour les ordinateurs d'apprendre sans être explicitement programmé » [5] et de Tom Mitchell en 1997 qui a défini l'apprentissage automatique comme « le processus consistant à apprendre à un ordinateur à effectuer une tâche particulière en améliorant sa mesure de performance avec l'expérience » [5].

ML est un domaine d'étude interdisciplinaire qui rassemble des techniques issues des domaines de l'informatique, des statistiques, des mathématiques et des sciences cognitives, notamment la biologie, la psychologie et la linguistique, pour n'en citer que quelques-unes [5]. Alors que l'idée d'apprendre à partir des données circule dans la communauté universitaire depuis plusieurs décennies, son entrée dans l'industrie technologique traditionnelle a commencé au début des années 2000 [5]. Cette croissance a coïncidé avec l'augmentation des données énormes à la suite de l'explosion du Web alors que les gens ont commencé à partager des données sur Internet [5].

4 Méthodes d'apprentissage automatique

La puissance du ML est due à la qualité de ses algorithmes, qui ont été améliorés et mis à jour au fil des ans ; ceux-ci sont divisés en trois méthodes sont: l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement [6]. Dans cette section, Nous présenterons en détail ces méthodes.

4.1 Apprentissage supervisé (Supervised Learning)

L'apprentissage supervisé (ou classification) consiste à construire un modèle basé sur un jeu d'apprentissage et des labels (nom des catégories ou des classes) et à l'utiliser pour classer des nouvelles données [7]. Cette technique est utilisée dans plusieurs applications telles que la prédiction des pannes et la détection des opinions trompeuses dans les réseaux sociaux. Il existe plusieurs algorithmes et techniques utilisés pour la classification supervisée telles que : Classification Bayésienne, Réseau neuronaux, Forêts d'arbres décisionnels (Random Forest), Machine à vecteurs de support (SVM) [7].

Dans l'apprentissage supervisé, il est possible de diviser les problèmes en fonction de la nature des données. Si la valeur de sortie est catégorique, comme l'appartenance / non-appartenance à une certaine classe, il s'agit d'un problème de classification. Si la sortie est une valeur réelle continue dans une certaine plage, il s'agit d'un problème de régression [6].

4.2 Apprentissage non supervisé (Unsupervised Learning)

L'apprentissage non supervisé contient un seul ensemble de données, cette méthode exige que le système de façon autonome doit restructurer les informations au sein de l'ensemble de données en les regrouper en sous-ensembles, de sorte que la plupart des données similaires soient dans le même ensemble [8]. Contrairement à l'apprentissage supervisé, il n'y a pas d'informations sur les classes d'appartenance des exemples ou généralement sur la sortie correspondant à une certaine entrée [6].

Les moteurs de recherche sont un exemple d'application de ces algorithmes. Étant donné un ou plusieurs mots-clés, ils sont en mesure de créer une liste de liens liés à notre recherche.

4.3 Apprentissage par renforcement (Reinforcement Learning)

L'apprentissage par renforcement vise à créer des algorithmes capables d'apprendre et de s'adapter aux changements environnementaux. Cette technique de programmation est basée sur le concept de réception de stimuli externes en fonction des choix de l'algorithme. Un choix correct impliquera une prime tandis qu'un choix incorrect entraînera une pénalité [6]. Le but du système est bien entendu d'obtenir le meilleur résultat possible [6].

Exemple : L'apprentissage par renforcement a tendance à bien fonctionner sur des jeux comme les échecs, où la récompense peut faire gagner le match. Dans ce cas, un certain nombre d'actions doivent être prises avant que la récompense soit atteinte [9].

5 Réseaux de neurones artificiels (Artificial Neural Network) ANN

Le réseau de neurones artificiels est un modèle spécial d'apprentissage automatique dont le fonctionnement est inspiré des neurones biologiques et qui s'appuie également sur les méthodes statistiques [4]. Une variété de tâches telles que la reconnaissance d'image, la reconnaissance vocale, la traduction automatique ainsi que le diagnostic médical utilisent ces ANN. Un avantage important d'ANN est le fait qu'il apprend à partir des exemples d'ensembles de données. Avec les ANN, on peut améliorer les techniques d'analyse de données existantes en raison de leurs capacités prédictives avancées. Il existe plusieurs types de réseaux neuronaux, dont chacun vient avec ses propres cas d'utilisation spécifiques et niveaux de complexité [4], les deux types les plus importants d'ANN sont [4]:

- **Réseau neuronal FeedForward (RNFF):** le flux d'informations n'a lieu que dans une seule direction. C'est-à-dire que le flux d'informations se voyage de l'entrée vers la sortie.

- **Réseau neuronal FeedBack (RNFB) :** les données peuvent circuler dans plusieurs directions.

Chaque neurone a son propre état interne interprété par la fonction d'activation. Il envoie son activation aux autres neurones sous forme de signaux. La connexion entre les neurones est réalisée via des liens orientés et pondérés [4].

[4] https://data-flair.training/blogs/artificial-neural-networks-for-machine-learning.

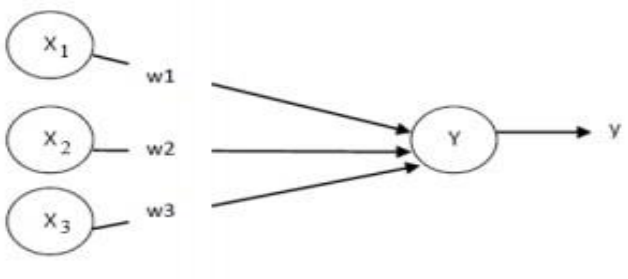


Figure 1 : Neurone artificiel avec une seule sortie [10].

Les Xi sont des valeurs numériques qui représentent soit les données d'entrée, soient les valeurs sorties d'autres neurones, ils ont comme valeurs de sortie $x1$, $x2$ et $x3$. Les poids Wi sont des valeurs numériques qui représentent soit la valeur de puissance des entrées, soit la valeur de puissance des connexions entre les neurones. Il existe des opérations qui se passent au niveau du neurone artificiel. Le neurone artificiel fera un produit entre le poids (w) et la valeur d'entrée (x), autrement dit la valeur d'entrée de neurone Y est : $\boldsymbol{y} = w1x1 + w2x2 + w3x3$, puis ajouter un biais (b), le résultat est transmis à une fonction d'activation (f) qui ajoutera une certaine non-linéarité (come l'exemple de la **figure 1**) [11].

6 Fonctions d'activation

Après que le neurone a effectué le produit entre ses entrées et ses poids, il applique également une non-linéarité sur ce résultat. Cette fonction non linéaire s'appelle la fonction d'activation.

La fonction d'activation est une composante essentielle du réseau neuronal. Ce que cette fonction a décidé si le neurone est activé ou non.

6.1 Types de fonction d'activation

6.1.1 La fonction Sigmoïde

La fonction sigmoïde illustrée à la **figure 2** est une fonction non linéaire qui ramène (ou écrase) les activations dans une plage de 0 et 1. Cela ramène les grands nombres négatifs et positifs à 0 et 1, respectivement. Les neurones commencent généralement à se déclencher lorsque la sortie de la fonction est supérieure à un seuil de 0,5 [5].

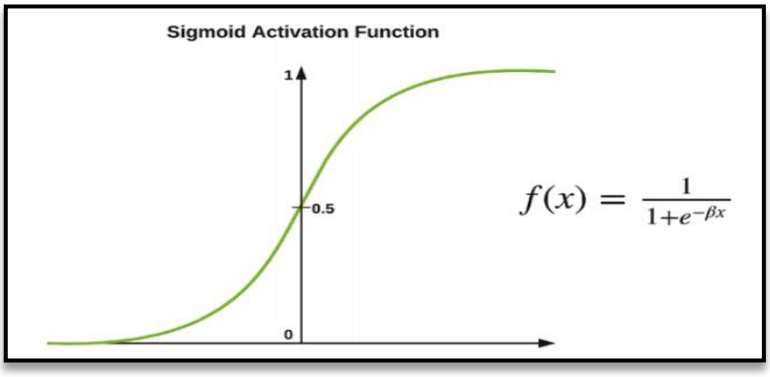


Figure 2 : Représentation graphique de la fonction d'activation Sigmoïde [5].

6.1.2 La fonction Tangente hyperbolique (Tanh)

La tangente hyperbolique illustrée à la **figure 3** améliore la fonction sigmoïde en limitant sa sortie dans une plage de −1 et 1. Ainsi, bien qu'elle souffre toujours du problème du gradient explosant et s'évanouissant, ses sorties sont maintenant centrées sur zéro. D'après la formule, le lecteur remarquera que tanh n'est qu'une fonction sigmoïde mise à l'échelle [5].

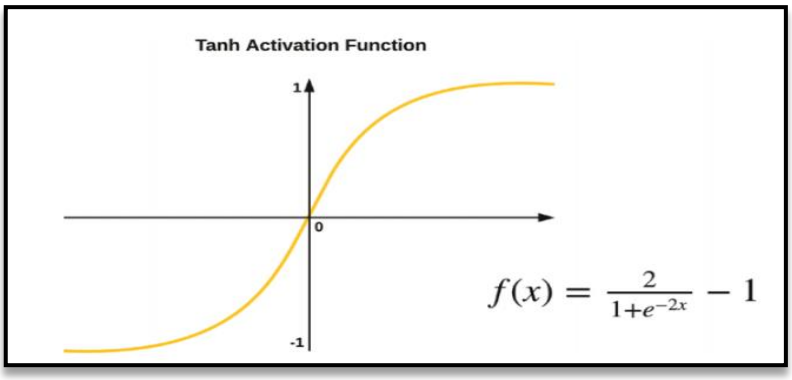


Figure 3 : Représentation graphique de la fonction d'activation Tanh [5].

6.1.3 La fonction Unité linéaire rectifiée (ReLU)

L'unité linéaire rectifiée ou la fonction d'activation ReLU est illustrée à la **figure 4** et fonctionne en réglant l'activation sur 0 pour les valeurs x inférieures à 0 et une pente linéaire de 1 lorsque les valeurs x sont supérieures à 0 [5].

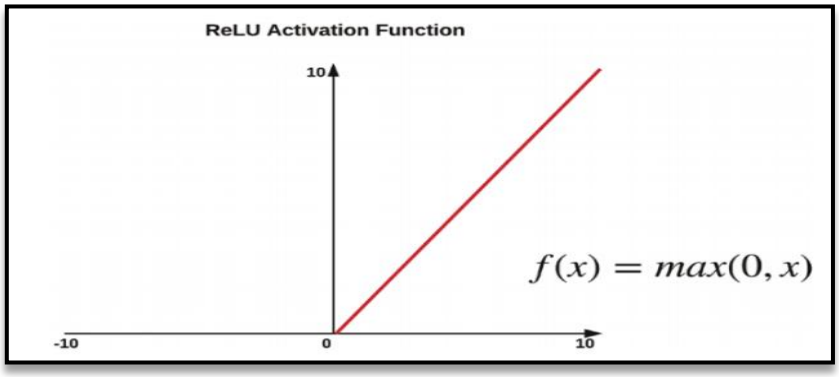


Figure 4 : Représentation graphique de la fonction d'activation ReLU [5].

ReLU offre une amélioration considérable sur les fonctions d'activation Tanh et sigmoïde en atténuant considérablement le problème de gradient de disparition et d'explosion. Cependant, certains gradients peuvent encore disparaître lors de la rétro-propagation avec un taux d'apprentissage élevé. Cependant, avec un rythme d'apprentissage bien défini, nous ne devrions pas avoir de problème [5].

6.1.4 La fonction Softmax

Softmax est utilisé comme fonction d'activation pour les problèmes de classification multi-classes où l'appartenance à une classe est requise sur plus de deux étiquettes de classe. Pour un vecteur réel arbitraire de longueur K, Softmax peut le compresser en un vecteur réel de longueur K avec une valeur dans l'intervalle [0, 1], et la somme des éléments dans le vecteur est 1 [5].

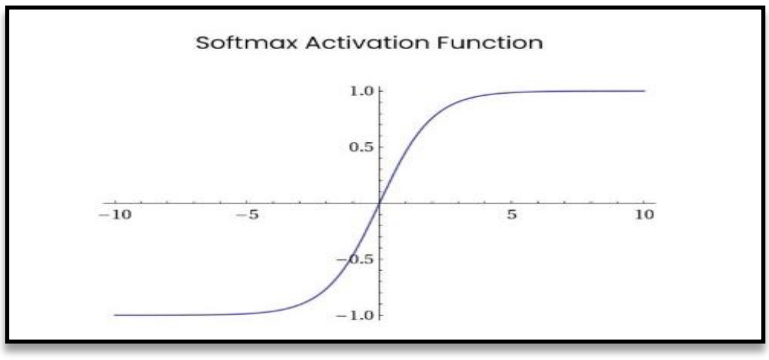


Figure 5 : Représentation graphique de la fonction d'activation Softmax [5].

Softmax est différent de la fonction max normale: la fonction max ne produit que la plus grande valeur et Softmax garantit que les valeurs plus petites ont une probabilité plus petite et ne seront pas rejetées directement [5].

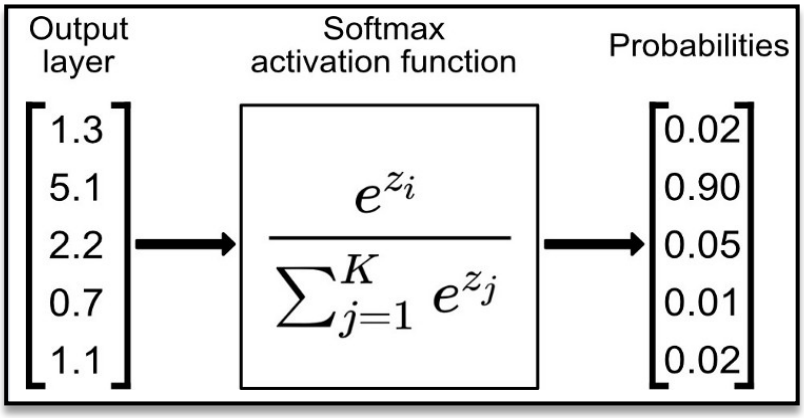


Figure 6 : Le processus de la fonction d'activation Softmax [5] **.**

La fonction Softmax pour la dernière couche de sortie, elle donne une probabilité pour chaque classe en attribuant une classe à chaque neurone (voir **figure 6**).

7 Fonction de perte

La fonction de perte est une méthode d'évaluation de la qualité d'un algorithme modélisant un ensemble de données. Si les prédictions sont totalement erronées, la fonction de perte affichera un nombre plus élevé. S'ils sont assez bons, cela produira un nombre inférieur [6].

Les fonctions de perte sont principalement classées en deux catégories différentes, à savoir la perte de classification et la perte de régression. La perte de classification est le cas où l'objectif est de prédire la sortie des différentes valeurs catégorielles, par exemple si nous avons un ensemble de données d'images manuscrites et que le chiffre à prédire se situe entre (0-9) [7].

Alors que si le problème est la régression comme prédire les valeurs continues par exemple, si vous devez prédire les conditions météorologiques [7].

[5] https://ichi.pro/fr/fonctions-d-activation-tout-ce-que-vous-devez-savoir-58662895551070
[6] https://algorithmia.com/blog/introduction-to-loss-functions

[7] https://www.analyticssteps.com/blogs/what-are-different-loss-functions-used-optimizers-neural-networks

7.1 Fonctions de perte de classification (Classification loss functions)

7.1.1 Entropie croisée binaire (Binary Cross-Entropy)

C'est une fonction de perte par défaut pour les problèmes de classification binaire. La perte d'entropie croisée calcule les performances d'un modèle de classification, qui donne une sortie d'une valeur de probabilité comprise entre 0 et 1 [8].

7.1.2 Entropie croisée multi-classes (Multi-class Cross-Entropy)

Dans ce cas, les valeurs cibles sont dans l'ensemble de 0 à n c'est-à-dire {0,1,2,3…n}. Il calcule un score qui prend une différence moyenne entre les valeurs de probabilité réelles et prévues, et le score est minimisé pour atteindre la meilleure précision possible. L'entropie croisée multi-classe est la fonction de perte par défaut pour les problèmes de classification multi-classe [8].

8 L'apprentissage profond (Deep Learning) DL

L'apprentissage profond est un sous-domaine du l'apprentissage automatique qui est un ensemble d'algorithmes inspirés de la structure et de la fonction du cerveau humain [9]. Ce type de système permet de traiter d'énormes quantités de données (big data) pour obtenir des relations et des modèles que les humains ne peuvent souvent pas détecter ou observer [12]. Il englobe l'apprentissage à différents niveaux de représentation et d'intangibilité qui aident à comprendre l'information, les images, les sons et les textes [10].

DL est un terme pour renommer les réseaux de neurones ou utiliser pour désigner des réseaux de neurones avec des couches successives (profondes). De plus, c'est un système qui s'apprend via les réseaux de neurones sans être dirigé par l'homme [13].

DL utilise des couches de neurones mathématiques pour traiter les données, identifier la parole et reconnaître les objets. Les données sont transmises par chaque couche, la sortie de la couche précédente accordant l'entrée à la couche suivante. La première couche d'un réseau est appelée couche en entrée et la dernière qui donne la réponse de classification est la couche en sortie. Les couches intermédiaires sont appelées couches cachées, et chaque couche du

[8] https://www.educba.com/loss-functions-in-machine-learning/
[9] https://openclassrooms.com/fr/courses/6417031-objectif-ia-initiez-vous-a-lintelligence-artificielle/6823506-apprehendez-le-deep-learning-ou-lapprentissage-profond

réseau est formée par un algorithme simple et uniforme qui englobe une sorte de fonction d'activation [10].

La **figure 7** représente un neural network avec 1 couche en entrée, 2 couches cachées et 1 couche de sortie.

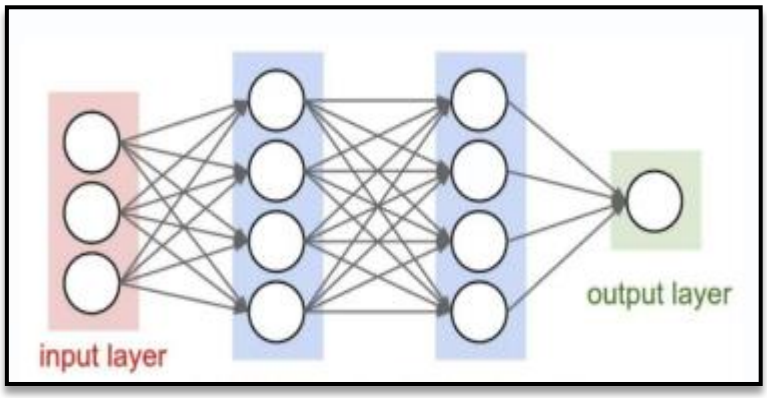


Figure 7: Neural network avec deux couches cachées [14].

8.1 La nécessité d'utiliser DL

Actuellement, DL est appliqué dans presque tous les domaines. La popularité de DL est due à sa précision, il a atteint des niveaux de précision plus élevés que d'autres algorithmes pour des problèmes de données complexes tels que le traitement du langage naturel (NLP). La capacité de DL à fonctionner de manière exceptionnelle a atteint des niveaux où les machines peuvent surpasser les humains, comme dans le cas de la détection de fraude [15].

Les performances des méthodes traditionnelles de ML avaient présenté de meilleures performances avec un minimum de données. Après avoir franchi le seuil, les performances des méthodes de ML traditionnelles deviennent stables, au contraire, les performances des méthodes DL augmentent avec l'augmentation de la quantité de données [16] comme le montre la **figure 8**.

[10] https://www.nucleodoconhecimento.com.br/administration-des-affaires/apprentissage-approfondi

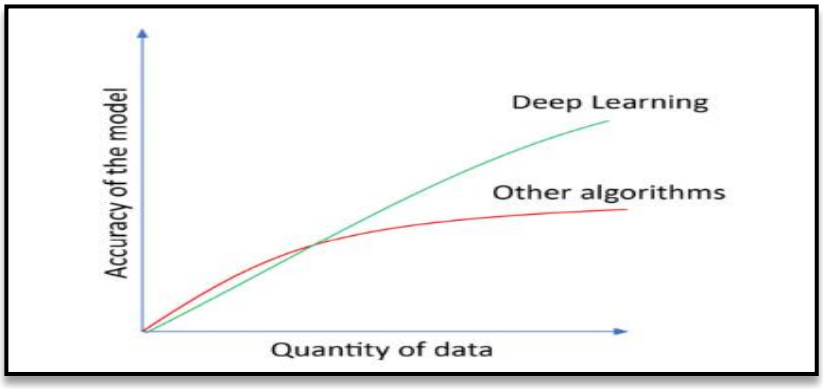


Figure 8: La performance des méthodes DL et autres méthodes en fonction de la quantité de données [15].

8.2 Les applications du l'apprentissage profond

Le DL a de nombreuses applications réussies dans de nombreux domaines, tels que [17] :

- **Traduction automatique :** Il s'agit d'une tâche dans laquelle des mots, expressions ou phrases donnés dans une langue sont automatiquement traduits dans une autre langue. La traduction automatique existe depuis longtemps, mais DL permet d'obtenir les meilleurs résultats dans deux domaines spécifiques :
  - Traduction automatique de texte.
  - Traduction automatique d'images.
- **Reconnaissance d'image :** Son objectif est de reconnaître et d'identifier les personnes et les objets dans les images, ainsi que de comprendre le contenu et le contexte. La reconnaissance d'image est déjà utilisée dans plusieurs secteurs tels que les jeux, les médias sociaux, le tourisme, etc. Cette tâche nécessite la classification des objets d'une image parmi un ensemble d'objets connus auparavant. Une variante plus complexe de cette tâche, appelée détection d'objet, consiste à identifier spécifiquement un ou plusieurs objets dans la scène de l'image et à dessiner un cadre autour d'eux.
- **Analyse des sentiments du texte :** De nombreuses applications ont des systèmes de révision basés sur des commentaires intégrés à leurs applications. La recherche sur le traitement du langage naturel et les réseaux de neurones récurrents ont parcouru un long chemin et il est maintenant tout à fait possible de déployer ces modèles sur le texte de votre application pour extraire des informations de niveau supérieur. Cela

peut être très utile pour évaluer la polarité sentimentale dans les sections de commentaires.

8.3 Couche Dropout

Le Dropout est une technique de régularisation dans les modèles de réseaux de neurones, proposée par Srivastava et ses collègues [18] afin de prévenir le sur-apprentissage. Dropout fournit une approximation peu coûteuse à la formation et à l'évaluation d'un ensemble de réseaux de neurones exponentiellement nombreux. Il est considéré comme plus efficace que d'autres régularisations standard qui sont peu coûteuses en calcul [18].

Plus précisément, dropout forme l'ensemble composé de tous les sous-réseaux qui peuvent être formés en supprimant les unités de non-sortie d'un réseau de base sous-jacent. Nous pouvons effectivement supprimer une unité d'un réseau en multipliant sa valeur de sortie par zéro [18].

Jusqu'à présent, le Dropout reste la méthode d'ensemble implicite la plus largement utilisée [18].

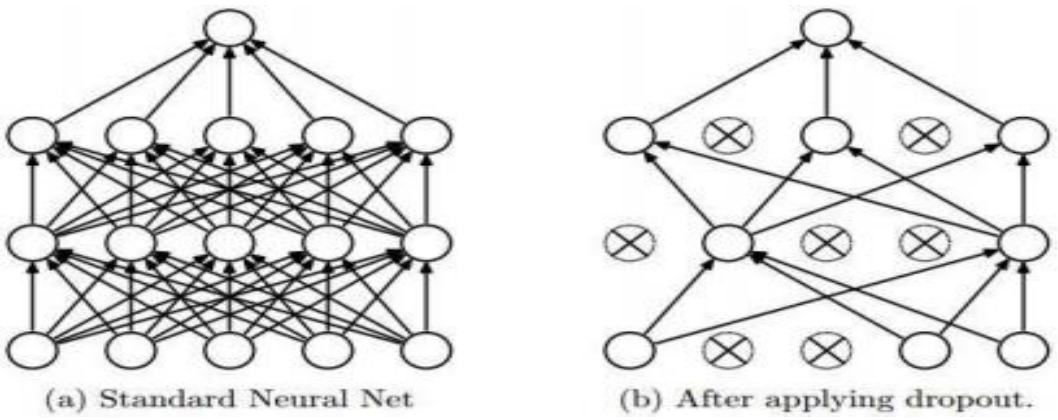


Figure 9 : Réseau avant et après dropout [19].

La **figure 9.a** représente un réseau neuronal où toutes les neurones sont activés tandis que la **figure 9.b** représente un réseau neuronal avec des neurones désactivés.

8.4 Perceptron

Le perceptron est un réseau de neurones composé de seulement un neurone, qui prend en entrée n données. Chacune de ses entrées i est pondérée par un poids noté wi. Le neurone peut prendre les états "1" ou "0" (respectivement positif ou négatif) en fonction de ses entrées

pondérées et d'un biais noté β ∈ R. Cet état représente la sortie du modèle. Il est donc possible de représenter le perceptron comme une fonction paramétrique fθ : {0, 1} n → {0, 1} avec θ l'ensemble de ses paramètres, c'est-à-dire le biais β et les poids w = (w1, . . . , wn) [20].

Dans la **figure 10** nous présentons un exemple de perceptron avec 3 données d'entrés (x1, x2, x3) sont pondérées par des poids mentionnées successivement w1, w2, w3 et une seule sortie y.

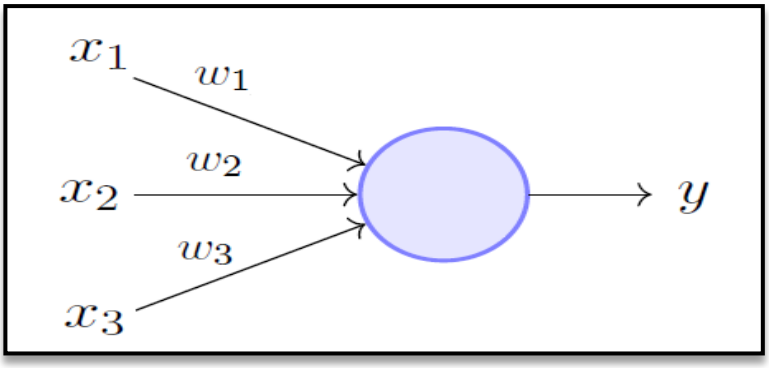


Figure 10 : Représentation d'un perceptron [21].

8.5 Perceptron multicouche (Multi layer Perceptron) MLP

Le perceptron multicouche est composé de neurones interconnectés, donc est un réseau neuronal. Spécifiquement, il s'agit d'un Réseau neuronal FeedForward, car il existe une direction au flux de données via le réseau (pas de connexions récurrentes de cycles). Ils relient plusieurs perceptrons (communément appelés neurones) ensemble dans un réseau, les neurones qui prennent la même entrée sont regroupés en une couche de perceptrons. Un MLP doit contenir une couche d'entrée et une de sortie et au moins une couche cachée. En outre, les couches sont également entièrement connectées (FC), ce qui signifie que la sortie de chaque couche est connectée à chaque neurone de la couche suivante. En d'autres termes, un paramètre de poids est appris pour chaque combinaison de neurone d'entrée et de neurone de sortie entre les couches [13]. La **figure 11** représente un exemple d'un perceptron de type multicouche avec une couche d'entrée des cinq données (x1, x2, x3, x4 et x5), une couche de sortie de deux sorties (y1 et y2) et trois couches cachées.

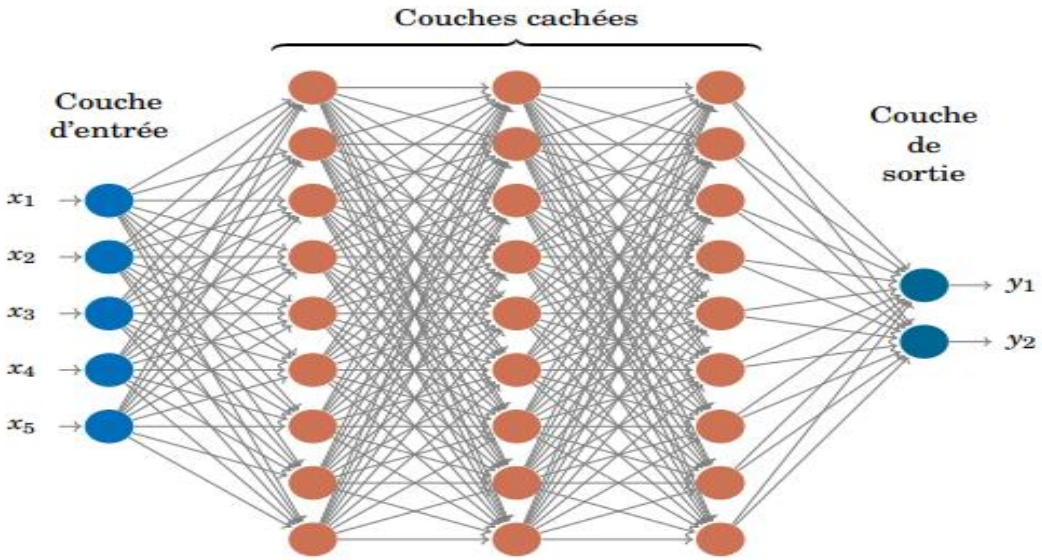


Figure 11 : Représentation d'un perceptron multicouche [22].

8.6 Réseaux de neurones convolutifs (Convolutional neural networks) CNN

Les réseaux de neurones convolutifs sont une classe de réseaux de neurones qui se sont avérés très efficaces dans les domaines de la reconnaissance d'images. Dans la plupart des cas, ils sont donc appliqués au traitement d'images [11]. Ils ont montré des performances exemplaires dans la segmentation, la classification et la détection. La caractéristique intéressante de CNN est sa capacité à exploiter la corrélation spatiale ou temporelle dans les données [23].

Les CNN ont obtenu une adoption et un succès énormes dans les applications de vision par ordinateur, mais c'est principalement avec l'apprentissage supervisé par rapport à l'apprentissage non supervisé qui a attiré très peu d'attention [11].

Les CNN sont également connus comme l'application des neurosciences au ML. Ils emploient des opérations mathématiques connues sous le nom de « Convolution » ; qui est un type spécialisé d'opération linéaire [11].

Ce réseau est un excellent exemple de variation pour le MLP dans le traitement et la classification. C'est un algorithme de DL dans lequel il prend les entrées comme une image et met efficacement des poids et des biais à ses objets et enfin capable de différencier les images les unes des autres [11].

[11] https://vinodsblog.com/2018/10/15/everything-you-need-to-know-about-convolutional-neural-networks/

- ✓ Les CNN sont composés de trois types fondamentaux de couches [5] :
  - **Couche de convolution**
  - **Couche de Pooling**
  - **Couche de Fully Connected**

La figure ci-dessous exprime un exemple d'une architecture du CNN en mentionnant les trois couches fondamentales pour la classification des images d'animaux.

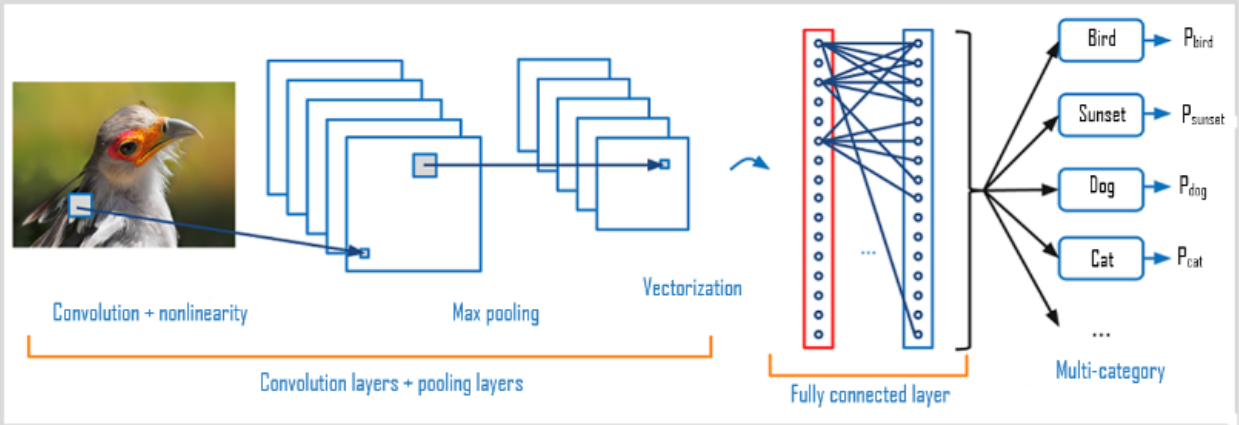


Figure 12 : Schéma de l'architecture du CNN pour une classification des images d'animaux [24].

8.6.1 Couche de convolution (Convolution layer) CONV

La couche de convolution est composée de filtres et de cartes de caractéristiques. Un filtre est passé sur les pixels de l'image d'entrée pour capturer un ensemble spécifique de caractéristiques (features) dans un processus appelé convolution. La sortie d'un filtre est appelée carte de caractéristiques (features map) [5].

La couche de convolution est l'élément central des CNN, elle compose au minimum leur première couche. Son objectif est de détecter la présence de caractéristiques (features) dans les images d'entrée. Cela est réalisé grâce à un filtrage par convolution qui consiste à faire glisser une fenêtre représentative de la caractéristique sur l'image d'entrée **(figure 13)** et à calculer le produit de convolution entre la caractéristique et chaque portion de l'image balayée [25].

Les considérations (paramètres) clés à prendre en compte lors de la conception d'une couche de convolution sont [5]:

- La taille du filtre (The filter size)
- La foulée du filtre (The stride of the filter)
- Le rembourrage pour l'entrée de la couche (The padding for the layer input)

La foulée du filtre détermine le nombre de pas de pixels que le filtre effectue lors du passage d'une activation d'image à une autre. Il est typique d'utiliser une foulée de 1, bien que cela puisse être augmenté pour les grandes images [5].

Parfois, le choix de la taille de notre filtre et la foulée (stride) sélectionnée peuvent ne pas diviser uniformément la taille de l'image d'entrée. Ainsi, pour éviter de perdre des informations sur les pixels puisque nous ne glissons pas au-delà du bord de l'image, une technique appelée zéro remplissage (zero padding) est utilisée pour remplir les bordures des pixels de l'image avec une couche définie de zéros. Cela permet au filtre de parcourir uniformément tous les pixels de l'image en incluant les zéros dans la convolution [5].

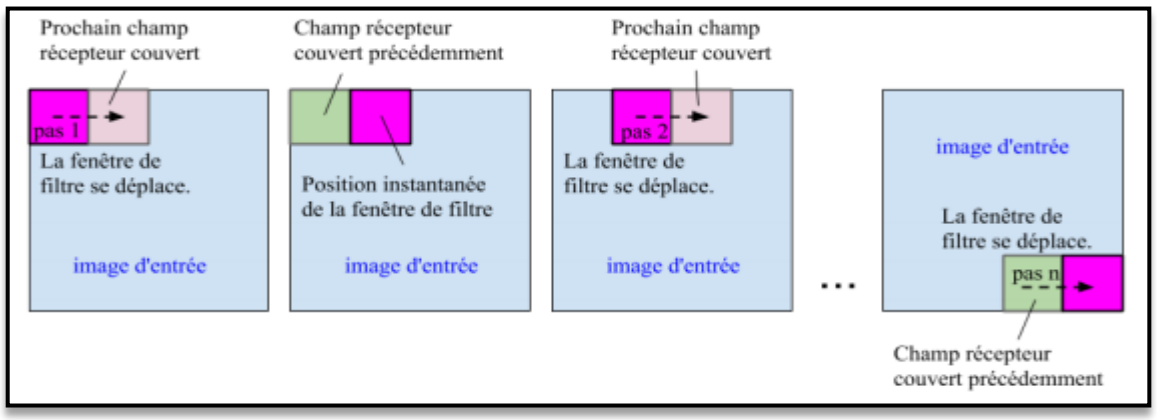


Figure 13 : Schéma du Glissement de la fenêtre de filtre sur l'image d'entrée [25].

Dans la figure ci-dessous nous citons un exemple sur le traitement du glissement d'un filtre dans l'image qui est représentée par une matrice. Cette figure compose d'une image d'entrée à gauche, un filtre au milieu et à droite l'image résultante (carte de caractéristiques) de processus de la couche convolution. Par exemple pour calculer la valeur finale de la position 3 du l'image d'entrée nous mettons le filtre exactement sur la position 3 de l'image d'entrée, puis nous calculons le produit entre eux jusqu'à la fin de la matrice appliquée et à la fin nous faisons la somme.

Résultat obtenu : 1 x 1 +0 x 0 + 0 x 1 + 1 x 0 + 1 x 1 + 0 x 0 + 1 x 1 + 1 x 0 + 1 x 1 = **4**

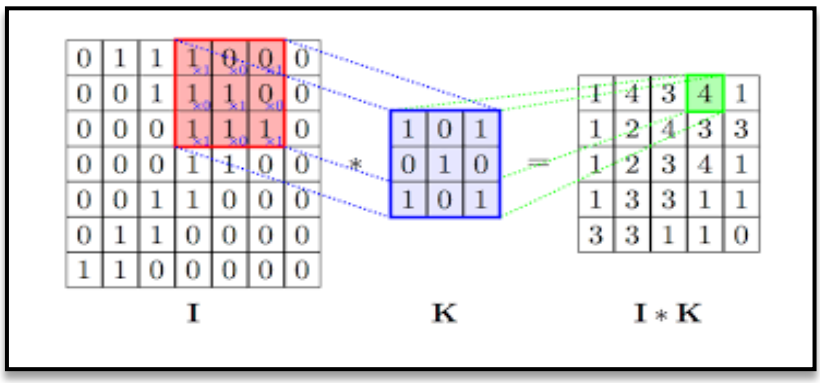


Figure 14 : Processus de convolution [26].

8.6.2 Couche de mise en commun (Pooling layer) POOL

Les couches de mise en commun suivent généralement une ou plusieurs couches de convolutions. L'objectif de la couche de mise en commun est de réduire ou de sous-échantillonner la carte des caractéristiques (features map) de la couche de convolution et extrait les caractéristiques essentielles seulement. La couche de mise en commun résume les caractéristiques (features) de l'image apprises dans les couches de réseau précédentes. Ce faisant, cela permet également d'éviter le sur-apprentissage du réseau [5].

L'avantage essentiel de la couche de pooling est sa capacité à injecter de l'invariance de localisation dans le réseau. L'invariance de localisation signifie que les caractéristiques peuvent être détectées par le réseau, peu importe où elles se trouvent sur l'image [5].

- ✓ En particulier, les types de pooling les plus populaires sont [12]:
  - **Max-pooling** : Son objectif est de sous-échantillonner une représentation d'entrée (image, matrice de sortie de couche cachée, etc.) en réduisant sa dimension en utilisant la valeur maximale.
  - **L'average pooling :** dont l'opération consiste à conserver à chaque pas, la valeur moyenne de la fenêtre de filtre.

Finalement, on obtient en sortie de cette couche de Pooling, le même nombre de cartes des caractéristiques qu'en entrée mais considérablement compressées.

[12] https://datascientest.com/convolutional-neural-network

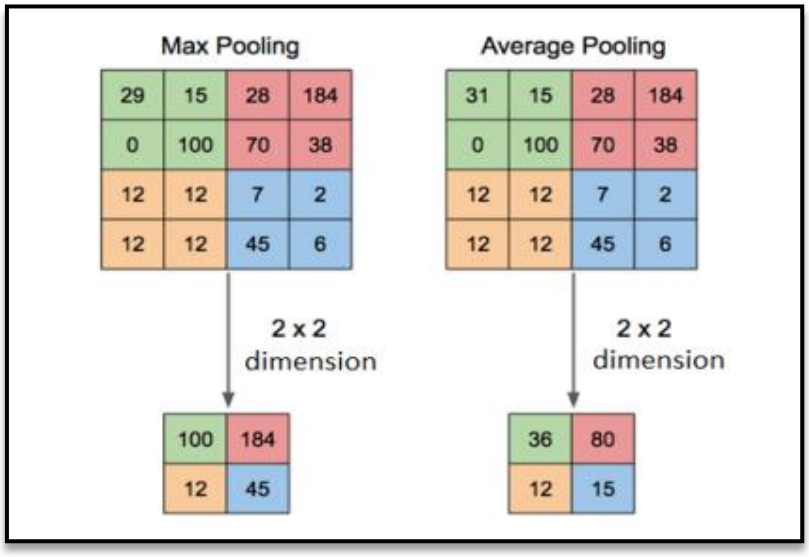


Figure 15 : un exemple de Max Pooling et Average Pooling avec une taille de filtre de 2x2 pixels à partir d'une entrée de 4x4 pixels [27].

Dans la **figure 15** :

- **Max pooling (à gauche) :** chaque case correspond à la valeur maximum du carré d'entrée de la même couleur, ex de la case verte : max (29, 15,0, 100) = 100.
- **Average pooling (à droite) :** chaque case correspond à la moyenne du carré d'entrée de la même couleur, ex de la case rouge : (28+ 184+ 70+ 38) /4 = 80.

8.6.3 Couche entièrement connectée (Fully Connected layer) FC

La couche entièrement connectée est un réseau de neurones à action directe ou un perceptron multicouche. Dans tous les cas, le réseau FC est la dernière couche du CNN. Dans ce cas, une fonction d'activation est utilisée pour générer les probabilités qu'une entrée appartienne à une classe particulière [5].

Avant de passer une entrée dans le réseau FC, la matrice d'entrée (la sortie de la couche de convolution ou de la couche de pooling) devra être aplatie c'est-à-dire deviendra un vecteur de 1 dimension (1D). Par exemple, une matrice d'image 28 x 28 x 3 deviendra 2352 poids d'entrée plus un biais de 1 dans le réseau FC [5].

Pour plus de précision, nous spécifions les sous-catégories de la couche FC :

- **La couche d'entrée entièrement connectée (Fully connected input layer) :** transformer les sorties générées par les couches précédentes en un seul vecteur 1D (**figure 16**).
- **La couche entièrement connectée (Fully Connected Layer) :** Elle s'applique sur une entrée aplatie qui est la sortie de la couche d'entrée entièrement connectée. Dans les réseaux de neurones, ce sont les couches où toutes les entrées d'une couche sont connectées à chaque unité d'activation de la couche suivante. Cette couche applique des pondérations à l'entrée générée par l'analyse d'entités pour prédire une étiquette précise.
- **La couche de sortie entièrement connectée (Fully connected output layer)** : génère les probabilités finales pour déterminer une classe pour l'image.

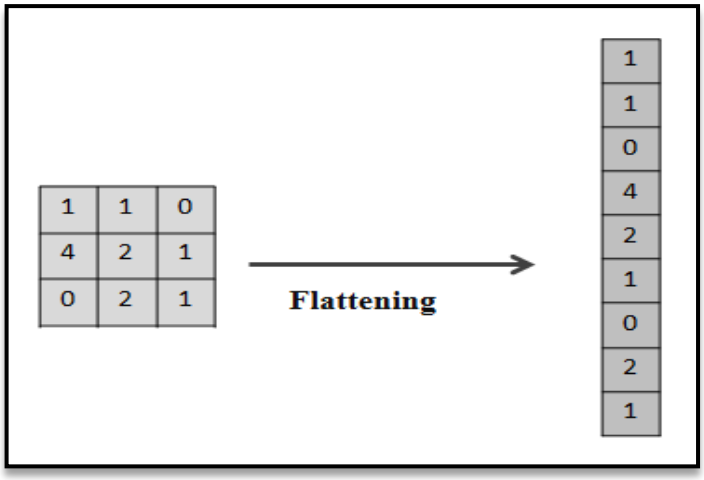


Figure 16 : Schéma de l'opération effectuée dans la couche d'entrée entièrement connectée (Flattening).

8.7 Auto-encoder

L'auto-encodeur est un type spécial du DNN (Deep Neural Networks) sans classe étiquetée, dont les vecteurs de sortie ont la même dimensionnalité que les vecteurs d'entrées. Il est souvent utilisé dans l'encodage de données. Il est entraîné pour tenter de copier son entrée vers sa sortie. En interne, il a une couche cachée h qui décrit un code utilisé pour représenter l'entrée. Le réseau peut être vu comme composé de deux parties : une fonction d'encodeur h = f (x) et un décodeur qui produit une reconstruction r = g(h). Cette architecture est présentée dans la **figure 17** [18].

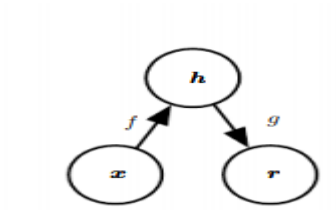


Figure 17 : La structure générale d'un auto-encodeur [18].

Les auto-encodeurs ne sont que des réseaux feedforward. Les mêmes fonctions de perte et types d'unités de sortie qui peuvent être utilisés pour les réseaux feedforward traditionnels sont également utilisés pour les auto-encodeurs [18].

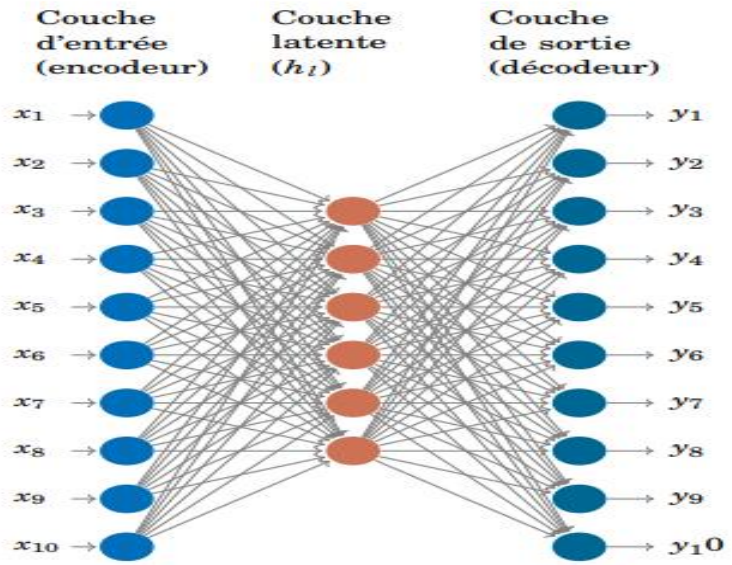


Figure 18 : Architecture d'un auto-encodeur [22].

- Un auto-encoder se compose de trois composants [13] (**figure 18**):

  Encodeur : L'encodeur est un réseau de neurones qui encode ou compresse les données d'entrée dans une représentation de l'espace latent. Les données compressées ont généralement l'air déformées, rien à voir avec les données d'origine.

  Représentation de l'espace latent : c'est un réseau de neurone qui contient la représentation réduite de l'entrée qui alimente le décodeur.

  Décodeur : le décodeur est également un réseau de neurone qui décode ou reconstruit les données encodées (représentation de l'espace latent) à leur dimension d'origine. Les données décodées sont une reconstruction avec perte des données d'origine.

[13] https://medium.com/@birla.deepak26/autoencoders-76bb49ae6a8f

✓ Il y a 4 paramètres que nous devons définir avant d'entraîner un auto-encodeur [14]:

Taille du code : nombre de nœuds dans la couche intermédiaire. Une taille plus petite entraîne plus de compression.

Nombre de couches : l'auto-encodeur peut être aussi profond que l'on veut.

Nombre de nœuds par couche : Le nombre de nœuds par couche diminue avec chaque couche suivante du codeur et augmente de nouveau dans le décodeur. De plus, le décodeur est symétrique du codeur en termes de structure de couche.

Fonction de perte : nous utilisons soit l'erreur quadratique moyenne (mean squared error) soit l'entropie croisée binaire (binary crossentropy). Si les valeurs d'entrée sont comprises dans la plage [0, 1], nous utilisons généralement l'entropie croisée binaire (binary crossentropy), sinon nous utilisons l'erreur quadratique moyenne (mse).

8.8 Réseaux de neurones récurrents (Recurrent neural networks) RNN

Les réseaux de neurones récurrents sont un autre schéma spécialisé d'architectures de réseaux de neurones, idéaux pour le traitement de données séquentielles 1D. Les RNN sont développés pour résoudre des problèmes d'apprentissage où les informations sur le passé (c'est-à-dire les instants/événements passés) sont directement liées à la réalisation de prédictions futures [5].

Un RNN est un réseau de neurones dont le graphe de connexion contient au moins un cycle [28].

Au cours des dernières années, un type de RNN est devenu plus connus grâce à ses excellentes performances sur des tâches aussi nombreuses que variées qui est les réseaux de neurones à base de cellules LSTM [28].

De tels exemples séquentiels se produisent fréquemment dans de nombreuses tâches du monde réel telles que la modélisation du langage où les mots précédents de la phrase sont utilisés pour déterminer ce que sera le prochain mot. Les RNN sont particulièrement adaptés

[14] https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798#3f72

aux séries chronologiques ou aux tâches séquentielles. Dans un problème séquentiel, il existe un cadre de bouclage ou de rétroaction qui relie la sortie d'une séquence à l'entrée de la séquence suivante [5]. En effet, la structure d'un RNN introduit un mécanisme de mémoire des entrées précédentes qui persiste dans les états internes du réseau et peut ainsi impacter toutes ses sorties futures [29].

Les RNN utilisant des unités LSTM et BLSTM ont des résultats de pointe dans de nombreuses tâches difficiles telles que la reconnaissance vocale, la traduction automatique, le sous-titrage d'images [30].

8.9 Mémoire à long court terme (Long Short Term Memory) LSTM

L'architecture de mémoire à long court terme a été proposée en 1997 par Hochreiter et Schmidhuber [25]. Les LSTM sont appelés gated car contrairement aux unités récurrentes de base, ils contiennent des composants supplémentaires appelés gâtes qui contrôlent le flux d'informations au sein de la cellule récurrente. Cela inclut le choix des informations à stocker dans la cellule et des informations à supprimer ou à oublier [5].

L'idée principale de l'architecture LSTM est de conserver une mémoire de toutes les entrées reçues par la couche cachée au fil du temps, en additionnant toutes les entrées (déclenchées) de la couche cachée dans le temps à une cellule mémoire [31].

Les composants du LSTM sont : la cellule mémoire, la porte d'entrée, la porte d'oubli et la porte de sortie. Il est important de noter que les composants des cellules LSTM sont tous des réseaux de neurones entièrement connectés [5].

9 Conclusion

Nous avons consacré ce chapitre à la présentation des notions de base de ML et DL, et ont concluons avec la description des principales architectures utilisées en DL dans le TAL, à savoir les CNN, les RNN, les LSTM ainsi que les Auto-encodeurs, qui promettent une révolution et une évolutivité dans notre domaine qui est le calcul du degré de dépression.

Notre travail s'appuyée sur le Traitement Automatique de la langue dans le domaine de l'analyse des sentiments plus précisément dans leur sous domaine qui est le calcul du degré de dépression à partir des réseaux sociaux, nous cherchons à proposer une solution plus performante à ce problème en incluant les approches du DL.

Chapitre 02 : Etat de l'art : La mesure de degré de la dépression

1 Introduction

Les sentiments exprimés dans les réseaux sociaux donnent une idée des émotions plus profondes des utilisateurs, ainsi les sentiments qui ont une signification négative peuvent indiquer une émotion négative.

L'analyse des sentiments est le domaine d'étude qui analyse les opinions, les sentiments, les attitudes et les émotions des gens. C'est l'un des domaines de recherche les plus actifs dans le traitement du langage naturel et a également été largement étudié dans l'exploration de données.

L'analyse des sentiments a récemment reçu beaucoup d'attention car les sentiments sont à la source de presque toutes les activités. Elle est très importante pour les entreprises, les organisations et la vie quotidienne d'une personne en générale dans la prise de décision. L'analyse des sentiments offre une technologie robuste à un certain nombre de domaines problématiques comme la détection précoce de la dépression et leur sous domaine mesurer la gravité des signes de dépression. Cela peut être attribué aux développements récents des médias sociaux et à la vitesse à laquelle les informations sont échangées entre les utilisateurs.

Dans ce chapitre, nous introduisons quelques concepts sur le traitement automatique du langage naturel, puis nous définirons ce que signifie l'analyse des sentiments et nous concentrerons sur notre problématique de mesurer la gravité des signes de dépression. Nous approfondissons notre problématique en partageant les différents travaux effectués de l'année passée des équipes participant à la conférence CLEF [15] eRisk2020 qui ont utilisé différentes approches pour résoudre ce problème.

2 Traitement Automatique De La Langue

L'analyse des sentiments dans les réseaux sociaux repose sur le traitement automatique de la langue naturelle pour analyser les émotions en ligne et déterminer les sentiments derrière la publication.

Bien que l'être humain soit capable d'analyser un discours ou un texte pour extraire et manipuler son contenu conceptuel, mais leur capacité est limitée à manipuler une petite quantité d'informations. Étant donné que la quantité d'informations disponibles en ligne dépasse

[15] https://early.irlab.org/

aujourd'hui de loin leur capacité de les traiter. Des méthodes capables de comprendre toutes ces informations sont devenues indispensables.

Le Traitement Automatique de la Langue Naturelle (TALN) est une discipline scientifique très récente. Né aux États-Unis vers 1949 [32], le TALN est dédié à la conception de méthodes et d'outils informatiques pour analyser la langue humaine [32].

Le traitement naturel du langage est loin d'être une tâche aisée. Le langage humain est par nature complexe et ses différentes règles sont difficiles à comprendre pour un ordinateur. Certaines de ces règles peuvent être très abstraites. Par exemple, lorsqu'une personne utilise une remarque sarcastique pour faire passer un message subtil. Il est presque impossible pour une machine actuelle de percevoir de telles nuances. Les difficultés rencontrées en TAL sont principalement de trois ordres, et ressortent soit de l'ambiguïté du langage, soit d'implicite contenue dans les communications naturelles et soit de la redondance [33].

3 L'analyse des sentiments

L'analyse de sentiments est un domaine en développement du traitement automatique du langage (TAL). Elle consiste à identifier le sentiment, l'opinion ou l'évaluation positive ou négatif exprimées à l'intérieur d'une unité informationnelle (i.e. mot, phrase, paragraphe ou document) [34]. L'analyse des sentiments est l'un des domaines de recherche les plus actifs en traitement automatique de langage naturel, Machine Learning, statistiques et linguistique depuis le début de l'année 2000 [35].

Dans la littérature, l'analyse des sentiments (sentiment analysis) est également appelée opinion Mining, opinion extraction, sentiment mining, subjectivity analysis [34].

Le but de l'analyse des sentiments est de définir des outils automatiques capables d'extraire des informations subjectives à partir du texte en langage naturel, telles que les opinions et les sentiments, afin de créer des connaissances structurées et utilisables par un système d'aide à la décision ou par un décideur [36].

3.1 Catégorisation des sentiments

Les phrases exprimant des opinions ou des sentiments sont généralement des phrases subjectives par opposition aux phrases objectives qui énoncent des faits. Cependant, les phrases

objectives peuvent également impliquer des sentiments positifs ou négatifs de leurs auteurs, car elles peuvent décrire des faits souhaitables ou indésirables [37].

La classification de subjectivité (Subjectivity classification) est la tâche qui distingue les phrases exprimant des informations objectives et les phrases exprimant des opinions subjectives [38].

La figure ci-dessous illustre un exemple de phrase objective et subjective.

Figure 19 : Exemple de phrase objective et subjective [38].

La classification de polarité (Polarity classification) est la tâche qui distingue les phrases qui expriment des polarités positives, négatives ou neutres [38].

La figure ci-dessous montre la classification de la subjectivité et de la polarité.

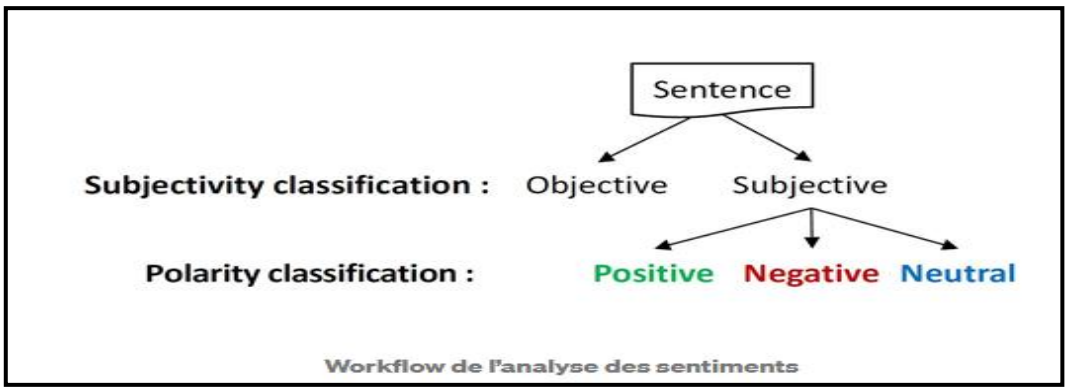


Figure 20: Classification de subjectivité et polarité [38].

3.2 Niveaux d'analyses

L'analyse du sentiment a trois niveaux d'analyse différents [37] : niveau du document, niveau de la phrase et niveau de l'aspect (représenté graphiquement dans la **figure 21**).

3.2.1 Niveau du document

La tâche à ce niveau est de déterminer l'opinion globale du document comme positif ou négatif. Ce niveau d'analyse suppose implicitement que chaque document exprime des opinions sur une seule entité (par exemple, un seul produit ou service) [39]. Cette approche ne convient pas dans le cas d'un document qui parle de différentes entités et contient des opinions sur différents objets comme dans les forums et les blogs [11].

3.2.2 Niveau de la phrase

Ce niveau consiste à déterminer la polarité de chaque phrase contenue dans un texte. L'hypothèse est que chaque phrase dans le texte exprime une opinion unique sur une entité unique [36].

3.2.3 Niveau des aspects

Effectue une analyse plus fine que les autres niveaux. Il repose sur l'idée qu'une opinion est constituée d'une polarité et une cible [36]. Dans ce cas, les traitements sont doubles : identifier d'abord l'entité et les aspects de l'entité en question, puis évaluer l'opinion sur chaque aspect [39]. Dans l'exemple suivant, la phrase « Samsung est très bon, mais il faut encore travailler sur la durée de vie de la batterie et les problèmes de sécurité » évalue trois aspects :

- Samsung est neutre.
- La durée de vie de la batterie est négative.
- La sécurité est négative.

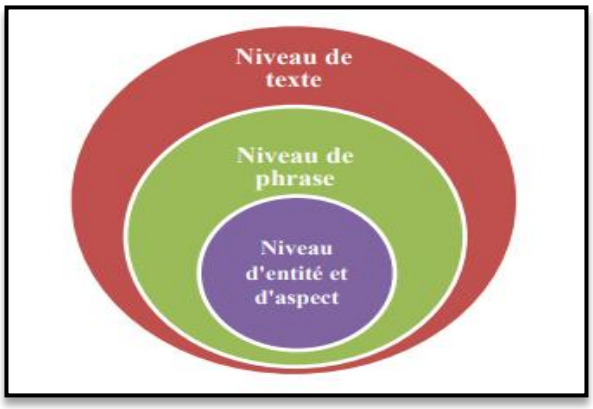


Figure 21 : Niveaux d'analyses [36].

3.3 Types d'analyse des sentiments

Pour comprendre les sentiments des gens, il existe de nombreux types d'analyses, y compris ceux qui se concentrent sur la classification de la polarité (positif, négatif, neutre), qui base sur analyse à grain fine des sentiments (très positif, positif, neutre, négatif et très négatif), qui détectent des émotions (en colère, heureux, triste, etc.) ou qui identifient des intentions (par exemple, intéressé, pas intéressé). Dans la section suivante, nous aborderons les types les plus importants.

3.3.1 Analyse à grain fine des sentiments (Fine-Grained Sentiment Analysis)

Il fournit une gradation d'opinion pour une réponse optimale. Par exemple, au lieu d'un sentiment positif, un sentiment neutre ou un sentiment négatif, il précis le sentiment en très positif, positif, neutre, négatif et très négatif. Cette analyse méticuleuse permet des réponses précises [16]. Les sentiments à grain fin sont courants dans la plupart des lieux de travail et vous pouvez les utiliser pour une note de 5 étoiles (par exemple : très positif peut représenter par 5 étoiles et très négatif peut représenter par 1 étoile) [17].

3.3.2 Détection d'émotion (Emotion Detection)

Ce type d'analyse des sentiments utilise des lexiques et l'apprentissage automatique pour identifier les types d'émotions heureuses, tristes ou en colère [17]. Détecter les sentiments des gens de cette manière est très difficile parce que les gens utilisent des mots avec des significations différentes. Par conséquent, une détection d'émotion inexacte peut conduire à une décision non fiable lors de l'analyse des vrais sentiments des personnes [18].

3.3.3 Analyse des sentiments à base d'aspects (Aspect-Based Sentiment Analysis ABSA)

Ce type analyse les aspects du texte pour déterminer le sentiment correspondant. En général, les aspects sont des attributs ou des composants d'une marque [15]. Par exemple quand vous parliez d'un haut-parleur ou d'un système de haut-parleurs sans fil. Vous pouvez analyser les sentiments de vos clients en leur demandant des commentaires sur la qualité de son, la connectivité et d'autres fonctionnalités, rendant ces appareils plus utiles et productifs pour les utilisateurs. Par conséquence il permet d'identifier les spécificités du produit [18].

3.3.4 Analyse des sentiments à base d'intention (Intent-Based Sentiments)

L'analyse de l'intention est une technique qui fournit un examen plus approfondi des sentiments. Elle détecte si les commentaires et textes collectés sont des plaintes, des suggestions, des demandes, etc. Cela aide à comprendre avec précision les réactions des personnes [16].

[16] https://tatvaminsights.com/what-is-sentiment-analysis/
[17] https://tatvaminsights.com/sentiment-analysis-positive-negative-and-neutral/
[18] https://www.cogitotech.com/blog/sentiment-analysis-types-how-it-works-why-difficult/

3.4 Les approches d'analyse des sentiments

L'analyse des sentiments utilise divers approches et algorithmes de traitement du langage naturel (TAL), que l'on peut classer comme suit:

3.4.1 Approche automatique

Cette approche utilise la technique de classification (technique d'apprentissage) pour classer le texte en des classes déférentes. Il existe principalement deux types de techniques d'apprentissage qui sont l'apprentissage supervisé et l'apprentissage non supervisé (ils sont détaillés dans le chapitre précédent) [19].

3.4.2 Approche à base de lexique

Cette approche est appelée aussi symbolique ou linguistique, jusqu'à maintenant, la plupart des études de l'analyse des sentiments se sont basées sur cette approche. Elle permet d'identifier la polarité d'un texte à l'utilisation de deux ensembles de mots, ceux qui expriment un sentiment positif et ceux qui expriment un sentiment négatif [8].

Le modèle compte dans le texte le nombre de mots positifs et le nombre de mots négatifs, si le nombre de mots positifs dépasse le nombre de mots négatifs alors le texte exprime un sentiment positif et si le contraire le texte exprime un sentiment négatif [8].

L'approche basée sur le lexique se divise en deux catégories :

A. Basé sur un corpus d'apprentissage

Les approches basées sur des corpus reposent sur l'orientation contextuelle des mots d'opinion [40]. Dans cette technique, un corpus de mots d'opinion est créé à partir d'un groupe de mots-clés de base pour un domaine spécifique. La taille augmente à chaque fois qu'en ajoute d'autres terminologies d'opinion [11].

B. Basé sur un dictionnaire

Elle s'appuie sur l'extraction de l'opinion puis la recherche dans un dictionnaire des synonymes et des antonymes prédéfinis, qui base sur l'orientation sémantique des mots tel que SentiWordNet qui est l'un des dictionnaires standards les plus utilisés de nos jours [41].

Le tableau ci-dessous montre les avantages et les inconvénients de l'approche basée sur l'apprentissage automatique et de l'approche basée sur un lexique :

Tableau 1 : Les avantages et les inconvénients de l'approche basée sur l'apprentissage automatique et de l'approche basée sur un lexique [42].

Approche	Approche automatique	Approche à base de lexique
Avantages	- Elle pourrait être transformé en ce que le domaine demande pour mieux travailler. - Un dictionnaire n'est pas nécessaire. - Donne des meilleurs résultats en terme de haute précision de classification.	- Ne demande aucune données d'entrainement ou des données étiquetées et ceci permet d'introduire moins d'opérations de calcul.
Inconvénients	- Elle peut être affectée par les variations de classes et aussi par l'effet des changements linguistiques. - Les classificateurs qui se sont entrainé sur un domaine spécifique, dans la plupart des cas ne fonctionnent pas avec un autre.	- Moins de capacité de classification en fonction du contexte ou du domaine. - Exigeait l'existence de ressources linguistiques puissantes qui ne sont pas toujours disponibles.

3.4.3 Approche hybride

Cette approche est appelée aussi classification semi-supervisée, elle combine les points forts des approches basées sur le lexique et des approches automatique [43]. La combinaison des deux approches précédentes a donné des résultats plus précis que les utilisent séparément [44].

3.5 Difficultés d'analyse des sentiments

Le domaine d'analyse des sentiments est très difficile. Dans ce qui suit nous citons quelques difficultés de cette procédure.

- Difficulté à extraire le sentiment en raison de l'ambiguïté de certains mots [45].
- Difficulté due à une mauvaise expression d'opinion, où une partie exprime des sentiments positifs et l'autre le contraire [44]. Par exemple, « L'histoire est intéressante mais les acteurs étaient mauvais. » Dans ce cas, la polarité du deuxième segment est opposée à la polarité du premier [46].
- Difficulté due à l'analyse de la phrase par « paquets de mots ». Les deux phrases suivantes contiennent les mêmes paquets de mots sans exprimer les mêmes sentiments. La première phrase contient un sentiment positif tandis que la seconde est négative. Exemple : « Je l'ai apprécié pas seulement à cause de ... », « Je l'ai pas apprécié seulement à cause de … » [44].
- Difficulté due au langage naturel pour l'analyse automatique de sentiments selon les contextes intentionnels, où l'expression d'une opinion n'est pas un sentiment réel. C'est le cas dans cette phrase : « Je croyais que la France était un beau pays. » [47].

3.6 Domaines d'application d'analyse des sentiments

L'utilisation de l'analyse des sentiments apparaît dans de nombreux domaines, et nous en mentionnerons quelques-uns ci-dessous.

- **Politique :** Grâce à l'analyse des sentiments, les décideurs de politique pouvant prenons l'avis des citoyens sur certaines politiques, afin de bénéficier de cette information pour améliore où créer une nouvelle politique qui convient avec les citoyens [19].
- **Prise de décision :** L'opinion et l'expérience des gens sont un élément très utile dans le processus de prise de décision [19].
- **Domaine médical :** Analyse l'opinion des médicine et des patient sur les médicaments et les services hospitalier, etc. Ainsi sur les documents de l'état de patient qui contient le diagnostic et la description du résultat d'examen... [19].

- **Domain éducation :** Développer le niveau pédagogique en analysant l'avis de l'étudiant à travers des méthodes pédagogiques et ça se permettre d'améliorer l'enseignement et l'apprentissage [19].
- **Marketing :** Du côté entreprises, permet au fournisseur plus de connaissances à propos des besoins des consommateurs et du côté client il peut donner son opinion, et s'inspirer des opinions d'autres clients pour l'aider à prendre sa décision et également comparer les produits avant de les acquérir [19].

4 Les troubles Dépressifs

La dépression (ou trouble dépressif) est une maladie psychique. Elle se manifeste par une tristesse durable, une perte d'énergie ou un abattement [19]. La dépression a des répercussions non seulement sur la santé en général et la survie mais aussi sur la vie familiale, les relations sociales et le travail. Un certain nombre de critères définissent la dépression, comme: l'humeur triste, une perte d'intérêt, un sentiment de découragement, des troubles du sommeil et de concentration, des perturbations de poids, un ralentissement psychomoteur ou une agitation, une fatigue chronique et des pensées récurrentes à la mort… [48].

La dépression peut survenir à tout âge, toucher tous les milieux socioprofessionnels et toutes les origines, il est plus fréquent chez l'adulte jeune et les femmes sont 2 fois plus touchées que les hommes [20].

4.1 Les facteurs de la dépression

Des situations et des événements de la vie, remontent parfois à l'enfance, peuvent favoriser la survenue d'une dépression :

- Relations perturbées avec les parents, expériences difficiles, etc.
- Décès d'une personne proche.
- Perte de son emploi.
- Séparation.

[19] https://www.ameli.fr/assure/sante/themes/depression-troubles-depressifs

[20] http://www.chikhineuropsychiatrie.com/depression-nerveuse/

- Conflit familial ou professionnel… [21]

5 Le degré de la dépression

Les troubles dépressifs surviennent souvent par étapes, sur une période de plusieurs semaines et parfois même des mois. Souvent, une personne touchée connaîtra plus d'un stade de dépression au cours de l'évolution de la maladie [22]. Les degrés de dépressions sont :

5.1 Une dépression minimale

La dépression minimale est un trouble de l'humeur qui n'est pas aussi grave que le trouble dépressif majeur. Par conséquent, seuls deux symptômes dépressifs doivent être présents, pendant deux semaines ou plus, pour un diagnostic de dépression minimale.

En règle générale, les personnes souffrant de dépression minimale présentent deux à quatre de ces symptômes, notamment une perte d'intérêt pour des activités ou des relations qui étaient auparavant agréables, des sentiments de tristesse et de désespoir, de la fatigue et une perte d'énergie, ainsi que de l'insomnie ou un sommeil excessif [23].

5.2 Une dépression légère

L'individu en souffre en raison de son incapacité à faire face aux événements qui l'entourent, tels que les pressions professionnelles et les problèmes familiaux, il a donc moins de sécrétions d'hormones antidépressives, et cela s'appelle la dépression réactive, et ses symptômes apparaissent à travers la souffrance du patient de : 1 - Détresse et colère constantes. 2 - Ennui et paresse à propos de certaines performances Tâches. 3 - Troubles du sommeil et de l'alimentation. 4- Évasion constante de la résolution de ses problèmes [24].

5.3 Une dépression modérée

Il survient principalement en raison de la présence de facteurs génétiques liés à un déséquilibre dans la chimie du cerveau qui empêche la sécrétion d'hormones antidépressives, et il apparaît après que l'individu a été exposé à un certain stress psychologique et nerveux, ou a été exposé à stress dans l'enfance qu'il ne pouvait pas oublier. Le patient souffre des choses qui sont déjà décrit dans la dépression légère en plus de: 1 - Tristesse extrême. 2 - Pensée à des choses pessimistes et se sentir sans valeur de la vie. 3 - Facilement excité nerveusement. 4 -Il

[21] https://www.ameli.fr/assure/sante/themes/depression-troubles-depressifs/comprendre-depression
[22] https://www.psychenet.de/ar/mental-disorders/mental-disorders-3/depression.html
[23] https://www.newportacademy.com/resources/glossary/minor-depression/

commence à se blesser en grattant légèrement sa peau avec des instruments tranchants, en décollant certaines de ses blessures. 5 - Des symptômes physiques apparaissent et il ressent des maux de tête, des douleurs abdominales, des douleurs musculaires, difficulté à respirer, en plus d'une peur obsessionnelle de la maladie et de la mort [24].

5.4 Une dépression majeure (Sévère)

L'une des étapes les plus dangereuses de la dépression, car elle survient à la suite d'une prédisposition génétique dans le système nerveux et ses symptômes apparaissent avec une exposition à un stress psychologique, de sorte que le patient n'est pas conscient de sa souffrance de dépression, et il entre dans épisodes d'isolement complet et de refus de la nourriture complètement, en plus de sa réflexion sur l'inutilité de la vie et la présence de chuchotements suicidaires Compulsif et en effet il accepte le suicide, souvent de manière dangereuse [24].

Dans ce type, le patient a besoin de traitements médicamenteux antidépresseurs avec des séances régulières de rythme cérébral, avec thérapie cognitive [24].

6 Détection précoce de la dépression à partit des réseaux sociaux

La dépression nécessite une détection précoce puisqu'une détection plus rapide, idéalement par l'identification de signes avant-coureurs, pourrait permettre de prévenir l'apparition de la maladie, en ayant recours à des traitements qui empêchent son développement. Les technologies de détection sont potentiellement utiles, pour cela la conférence d'eRisk dans sa première année de création en 2017 a mené une tâche exploratoire sur la détection précoce de la dépression à laquelle ont participé plusieurs équipes, qui consiste à découvrir les utilisateurs déprimés et non déprimés à partir ses textes écrits dans le réseau social Reddit, en utilisant divers algorithmes tels que l'apprentissage automatique et l'apprentissage en profondeur.

eRisk est un laboratoire CLEF sur la détection précoce des risques sur internet a été lancé en 2017 et s'est poursuivi jusqu'à cette année.

[24] https://www.elconsolto.com/psychiatric/psychiatric-news/details/

Reddit était la principale source de données pour les tâches expérimentales d'eRisk. Il s'agit d'une plateforme de discussions sur internet s'organisant autour de sujets postés par les internautes, regroupés dans différentes catégories [25].

Cette tache de détection précoce de la dépression a été effectuée par deux travaux dans notre département Informatique Blida1, le premier en 2017-2018 [49] et le deuxième en 2018-2019 [50] comme suit :

Dans le travail [49], les auteurs ont pris Twitter comme source de données, c'est l'ensemble de données qui est constitué de trois (3) parties : ensemble de données sur la dépression contient des tweets de personnes déprimées, ensemble de données non-dépression contient des tweets de personnes non déprimées et ensemble de données sur le candidat à la dépression contient des personnes potentiellement déprimées où ses tweets contiennent la chaîne « dépression ». Cette équipe a appliqué le prétraitement et une vectorisation basant sur le modèle word2vec. Ce travail repose sur la combinassions d'un modèle CNN avec le modèle MSA qui est un modèle BiLSTM à 2 couches avec un mécanisme d'attention permet au modèle de trouver l'importance relative de chaque mot dans l'expression en attribuant un poids ai à chaque annotation de mot, afin d'améliorer le processus d'extraction des caractéristiques et d'améliorer les performances du modèle. Le modèle a atteint une précision de 99% dans la détection de dépression. Ils ont également formé le modèle CNN et le modèle MSA séparément, les deux modèles ont obtenu une précision de 98%.

Dans l'étude [50] l'équipe a proposé quatre modèles d'apprentissage supervisé. Ils ont utilisé l'ensemble de donnée eRisk2019 pour l'apprentissage (Train data) contient 486 utilisateurs dont 83 sont déprimés et 403 sont non déprimés, et un autre ensemble de données eRisk2019 pour le test (Test data) contient 401 utilisateurs dont 52 déprimés et 349 non déprimés. Ils sont passés par un prétraitement et une vectorisation de données sur la base du modèle Glove. Ses modèles proposés sont les réseaux de neurones convolutifs (CNN), LSTM, BiLSTM et Naïve Bayes. Où Le modèle Naïve Bayes a eu 82% accuracy, 77% Recall, 77% F1-score, le modèle CNN a eu 86% accuracy, 79% Recall, 78% F1-score, le modèle LSTM a eu

[25] https://www.lesnumeriques.com/telecharger/reddit-29664

90% accuracy, 89% Recall, 84% F1-score et le modèle BiLSTM a eu 97% accuracy, 95% Recall, 92% F1-score.

7 Mesure du degré de la dépression à partir les réseaux sociaux

En 2019, la conférence CLEF eRisk présente pour la première fois un challenge qui vise à mesurer le degré de dépression à partir du fil des soumissions des utilisateurs comme une 2ème tache [51, 52].

Pour eRisk2020, la quatrième édition de ce laboratoire dans le cadre de la conférence CLEF. Le laboratoire avait deux tâches, la deuxième étant une continuation de la deuxième tâche de 2019. La tâche consiste à mesurer la gravité des signes de dépression. Pour chaque utilisateur, les participants ont reçu l'historique complet des publications de l'utilisateur (en une seule diffusion de données) et les participants ont dû remplir un questionnaire standard sur la dépression basée sur les preuves trouvées dans l'historique des publications. En 2020, les participants ont eu l'opportunité d'utiliser les données de 2019 comme données d'apprentissage (questionnaires remplis et soumissions des utilisateurs 2019, soit un ensemble de donnée composé de 20 utilisateurs contenant leurs publications) [53]. Les questionnaires sont dérivés du l'inventaire de dépression de beck (Beck's Depression Inventory) BDI [54], qui évalue la présence de sentiments comme la tristesse, le pessimisme, la perte d'énergie, etc. Le questionnaire contient 21 [53] questions (voir annexe). Les participants de la quatrième édition ont reçu un jeu de données avec 70 utilisateurs et on leur a demandé d'estimer les réponses de questionnaire BDI pour les 70 utilisateurs et de produire un fichier avec la structure suivante: nom d'utilisateur1 réponse1 réponse2 .... réponse21 nom d'utilisateur2 …. [53], en utilisant divers algorithmes.

Les paramètres qu'ont été pris en compte pour l'évaluation des résultats sont Average Hit Rate (AHR), verage Closeness Rate(ACR), Average Difference between overall depression levels (ADODL), Depression Category Hit Rate (DCHR) [52], ils seront détaillés dans le chapitre 4.

➢ Les travaux de recherches similaires

Dans cette partie nous présentons 5 travaux sur la mesure de gravité des signes de dépression des participants à la conférence CLEF eRisk2020.

Le tableau 2 rapporte les équipes participantes et les courses (Runs) pour lesquelles elles se sont soumises eRisk.

Tableau 2 : Le nombre de courses soumises pour chaque équipe [53].

Team	Runs
BioInfo@UAVR	1
RELAI	5
Prhlt-upv	4
iLab	3
USDB	4

Dans l'étude [55], **l'équipe BioInfo@UAVR** a utilisé l'ensemble de données eRisk2019. La première étape de l'approche pour aborder la tâche 2 consistait à prédire si un utilisateur était déprimé à l'aide du classificateur précédemment formé en 2019. Ensuite, ils ont conjugué le score de cette classification avec plusieurs modèles psycholinguistiques et comportementaux. Pour chaque catégorie, un score a été calculé pour chaque utilisateur comme une valeur normalisée du nombre d'occurrences des entités considérées pour chaque catégorie par rapport au nombre total d'occurrences des mêmes entités sur l'ensemble de données. Ces scores ont ensuite été normalisés à l'intervalle [0,3].

À cette fin, ils ont utilisé Empath, un cadre TAL pour calculer la polarité moyenne des écrits d'un utilisateur.

- Les utilisateurs déprimés ont tendance à utiliser de mots liés à soi (par exemple : je, moi-même, le mien) plus souvent dans leurs écrits.

- Utilisation de mots absolutistes.

- Mentions de mots liés aux troubles mentaux, (ex. : dépression, bipolaire, schizophrénie, psychotique).

- Utilisation des mots cri, culpabilité et leurs dérivés.

- Utilisation des mots sommeil, anxieux et leurs dérivés.

- Utilisation des mots irrités, fatigué et leurs dérivés.

Dans l'étude [56], **l'équipe USDB** a utilisé l'ensemble de données que CLEF eRisk a fournies en 2019 comme ensemble d'apprentissage (Train Data). Cette équipe a commencé par le prétraitement des données puis la vectorisation basée sur le modèle word2vec. Dans l'étape de construction du modèle ils ont exploité principalement deux types de réseaux neuronaux profonds, un réseau neuronal convolutif « CNN » et un réseau de neurones récurrent bidirectionnel à longue mémoire à court terme « BiLSTM ». Pour les deux modèles ils ont utilisé deux méthodes statistiques différentes pour la prédiction. Le max consiste à calculer pour une question par exemple « Sadness » la fréquence de chaque réponse générée pour choisir la réponse avec la plus grande occurrence. La suite consiste à calculer pour une question par exemple « Sadness » le nombre de suit de chaque réponse puis prendre la réponse qui a la plus grande suite.

Dans l'étude [57], **l'équipe iLab** a utilisé des classeurs basés sur BERT. Ils ont effectué trois éxécutions XLM-RoBERTa, RoBERTa-base et RoBERTa-base. Là où, la différence entre les trois exécutions est la taille des phrases dans la phase d'entraînement, dans la première et la deuxième exécution toutes les phrases sont de la même taille avec 128 jetons. Pour la troisième exécution, les phrases supérieures à 128 jetons ont été tronquées pendant la phase d'entraînement. Cette équipe a utilisé l'ensemble de donnée eRisk2019 pour l'apprentissage de ses modèles.

Dans l'étude [58], **l'équipe Prhlt-upv** a utilisé des modèles d'apprentissage automatique plus simples (SVM et régression logistique) à l'aide d'un ensemble de données eRisk2019. LogReg-features, le premier modèle utilisé était un modèle de régression logistique avec les caractéristiques basées sur le lexique représentées sous forme de vecteurs numériques. SVM-features Pour le deuxième modèle, ils ont utilisé un SVM avec noyau RBF, avec les mêmes caractéristiques de premier modèle. SVM-USE, le dernier modèle était un SVM avec noyau RBF et caractéristiques USE.

Dans [59], **l'équipe RELAI** aborde trois approches avec différentes exécutions. Elle a utilisé LDA avec deux exécutions différentes, une basée sur l'utilisateur et une basée sur la réponse. Contextualizer pour la deuxième approche, a utilisé deux exécutions, la première encode les documents ensemble (simultané) basé sur la réponse (answer-based, simultaneous), la seconde encode chaque document individuellement (parallèle) basé sur l'utilisateur (user-based, parallel). Ainsi qu'une approche basée sur une méthode de mise à l'échelle (Stylométrie), Cette approche se concentre sur le style d'écriture d'un document afin de caractériser son auteur. À cette fin, plusieurs caractéristiques linguistiques ont servi de représentations documentaires, telles que la longueur des mots et des phrases, la fréquence des mots et des caractères.

8 Discussion

Nous avons remarqué lors de notre revue de ces articles que chacun des participant a choisi une approche différente avec plusieurs méthodes et algorithmes, et tous les équipes ont utilisé l'ensemble de données eRisk2019 pour l'entrainement et l'ensemble de données eRisk2020 pour le test. Nous avons remarqué également qu'aucune équipe n'était en mesure d'obtenir les meilleurs résultats pour les quatre métriques d'évaluation. De plus, le sujet en lui-même est récent et traiter seulement par la conférence CLEF eRisk.

Le tableau suivant représente une comparaison entre les travaux présentés lors de la conférence CLEF eRisk2020 qui abordent la même problématique que nous traitons:

Tableau 3 : Tableau comparatif des travaux de mesurer de gravité de la dépression (Tache 2) eRisk2020.

EQUIPE	DATASET	ALGORITHME DE CLASSIFICATION	PERFORMANCES
		XLM-RoBERTa-base	**AHR : 36.73%** **ACR : 68.68%** **ADODL:81.07%** **DCHR : 27.14%**

iLab **[57]**	**eRisk2020**	**RoBERTa-base**	**AHR : 37.07%** **ACR : 69.41%** **ADODL : 81.70%** **DCHR : 27.14%**
		RoBERTa-base	**AHR : 35.99%** **ACR : 69.14%** **ADODL : 82.93%** **DCHR : 34.29%**
USDB **[56]**		**CNN_max**	**AHR : 34.97%** **ACR : 67.19%** **ADODL : 76.85%** **DCHR : 25.71%**
		CNN_suite	**AHR : 32.79%** **ACR : 66.08%** **ADODL : 76.33%** **DCHR : 17.14%**
		BiLSTM_max	**AHR : 34.01%** **ACR : 67.78%** **ADODL : 79.30%**

			DCHR : 22.86%
		BiLSTM_suite	**AHR : 33.54%** **ACR : 67.26%** **ADODL : 78.91%** **DCHR : 20.00%**
PRHLT-UPV **[58]**		**LogReg-features**	**AHR :34.01%** **ACR :67.07%** **ADODL :80.05%** **DCHR :35.71%**
		SVM-features	**AHR :34.56%** **ACR :67.44%** **ADODL :80.63%** **DCHR :35.71%**
		SVM-USE	**AHR:36.94%** **ACR:69.02%** **ADODL:81.72%** **DCHR:31.53%**

RELAI [59]		**LDA (answer-based)**	**AHR: 28.50%** **ACR: 60.79%** **ADODL: 79.07%** **DCHR: 30.00%**
		LDA (user-based)	**AHR: 36.39%** **ACR: 68.32%** **ADODL: 83.15%** **DCHR: 34.29%**
		Contextualizer (answer-based, simultaneous)	**AHR: 21.16%** **ACR: 55.40%** **ADODL: 73.76%** **DCHR: 27.14%**
		Contextualizer (user-based, parallel)	**AHR: 36.80%** **ACR: 68.37%** **ADODL: 80.84%** **DCHR: 22.86%**
		Stylometry (user-based)	**AHR: 37.28%**

			ACR: 68.37% **ADODL: 80.70%** **DCHR: 20.00%**
BioInfo@UAVR **[55]**		**Modèle basé sur psycholinguistiques et comportementaux**	**AHR : 38.30%** **ACR : 69.21%** **ADODL : 76.01%** **DCHR : 30.00%**

D'après le tableau de comparaison au-dessus, nous voyons que l'équipe BioInfo@UAVR avait un score plus élevé pour AHR avec 38.30%. Pour ACR, l'équipe iLab pour la deuxième exécution a un score de 69.41% en première place, et l'équipe BioInfo@UAVR et Prhlt sont proches d'elle avec des scores 69.21%, 69.02%, respectivement. ET pour ADODL l'équipe qui a eu le meilleur score était Relai avec 83.15%. Et pour la dernière mesure (DCHR) que l'équipe Prhlt qui a eu un meilleur score que les autres équipes, pour 3 exécutions avec le même score était de 35.71%.

D'un autre côté, nous voyons que la métrique ADODL pour toutes les équipes avait une haute performance, puis ACR et DCHR à la dernière place.

9 Conclusion

Dans ce chapitre, nous avons présenté les travaux liés à notre étude afin de proposer une approche qui donnera de meilleurs résultats.

Le chapitre suivant de ce mémoire décrit notre modèle proposé et les étapes que nous avons suivies pour le construire et le tester en détail.

Chapitre 03 : Conception des modèles pour calculer le degré de la dépression à partir des réseaux sociaux.

1 Introduction

Après avoir passé en revue les travaux connexes dans ce domaine, nous présentons dans ce chapitre notre architecture proposée pour calculer le degré de dépression à partir des postes de reddit des utilisateurs fournis par la conférence CLEF eRisk. Nous expliquons également en détail les parties qui composent notre architecture proposée (Préparation de données (Train data, Test data), Prétraitement et vectorisation, Apprentissage des modèles, prédiction des résultats).

2 Architecture proposée

Dernièrement, les réseaux de neurones se sont avérés très efficaces pour la classification de textes, et après une très vaste recherche nous n'avons pas trouvé d'études portant sur l'Auto-encoder dans la classification de textes et aucune des travaux connexes dans ce problème a utilisé l'Auto-encoder pour calculer le degré de dépression. Les résultats de ce type de classifieur dans le traitement des images sont excellents, pour cela l'originalité de notre travail porte sur la proposition de deux modèles le CNN et la combinaison de l'Auto-encoder avec CNN (Auto-encoder_CNN).

A la création de nos modèles, nous passons par quatre parties essentielles sont les suivantes :

Partie 1 : Préparation de données (Train data, Test data).

Partie 2 : Prétraitement et vectorisation.

Partie 3 : Apprentissage des modèles.

Partie 4 : prédiction des résultats.

Nous schématisons les étapes constituant notre architecture dans la figure ci-dessous **(figure22)**.

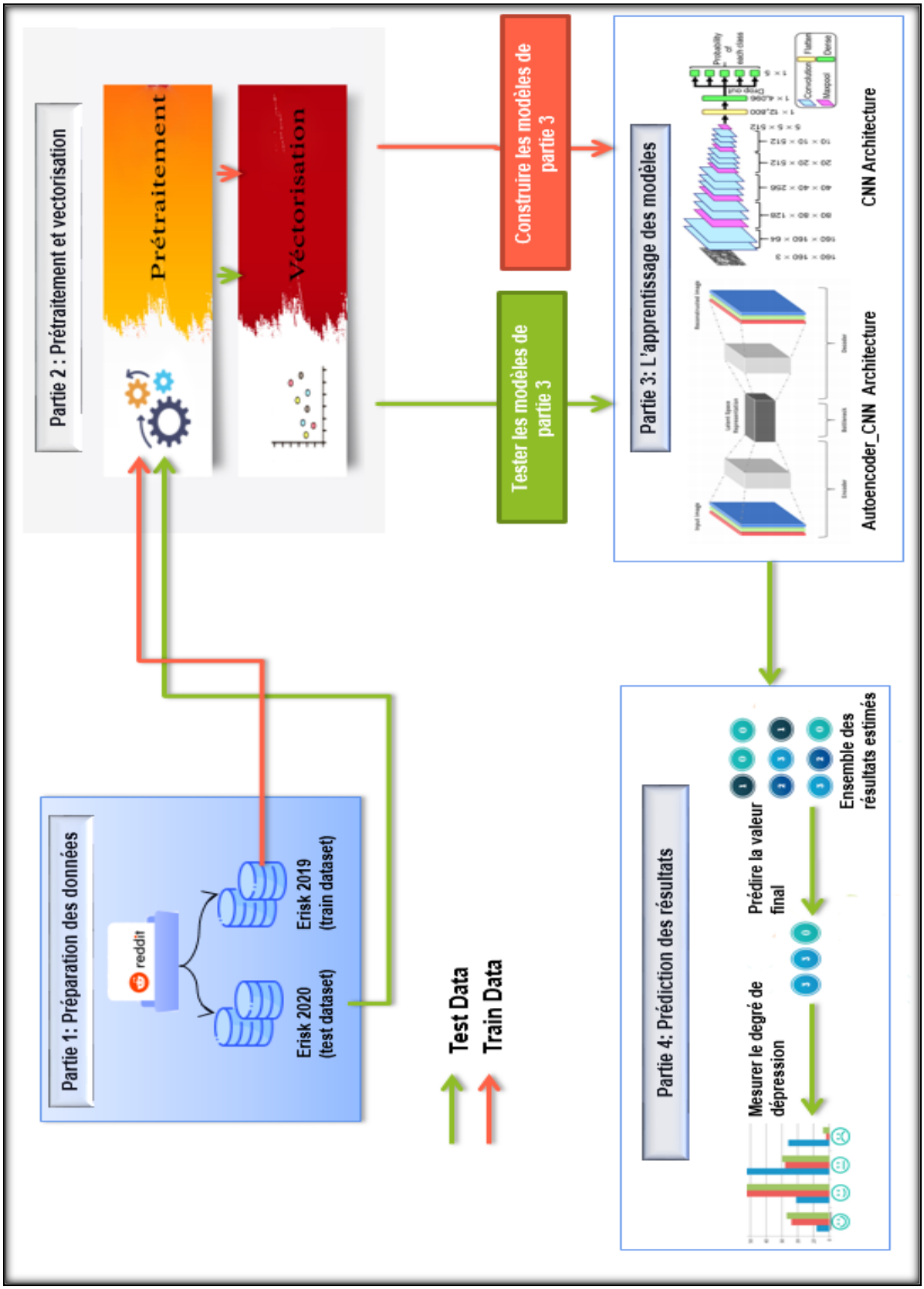


Figure 22 : Architecture proposée.

2.1 Partie 1 : Préparation de données (Train data, Test data)

Nous utilisons deux ensembles de données (dataset) fournis par eRisk, la première est celle de 2019 pour l'apprentissage des modèles (train dataset) et la deuxième celle de 2020 pour le test (test dataset). Plus de détails sur ces données seront présentés dans le chapitre 4.

Train Data

L'ensemble de données 2019 (train dataset) se compose de 20 fichiers XML nommé « Subject » ; il correspond à l'historique des publications des personnes dépressives sur Reddit et un fichier TXT contenant les 21 réponses réelles du questionnaire de « BDI ». Pour permettre utiliser l'ensemble de données 2019 dans l'entrainement - d'une façon fiable ; nous combinons les fichiers XML et le fichier TXT d'eRisk2019 et les regrouper dans un seul fichier CSV (**figure 23**).

ID	TEXT	Sadness	Pessimism	Past Failure	Loss of Pleas
subject1272	Just waking up every morning and talking to my gf. I really felt nothing inside unt	1	2	2	
subject1272	I upvote everything, because everyone deserves love	1	2	2	
subject1272	I would say being born, but Ill stick to an original answer and take myself back to	1	2	2	
subject1272	All star	1	2	2	
subject1272	Excuse me but why tf are ur eyes crossed?	1	2	2	
subject1272	Im sorry if Im dumb but do what?	1	2	2	
subject1272	Same as hes done: nothing	1	2	2	

Figure 23 : Structure d'un fichier d'apprentissage csv.

Test Data

Pour l'ensemble de données 2020 se compose de 70 fichier XML, nous avons extrait toutes les publications de chaque utilisateur et les mettre dans des fichiers CSV individuellement (**figure 24**).

TEXT
Is the Naval Academy's logo copyrighted?
Ah, muscle control, the kinds of things I take for granted as an adult.
Hangovers are a great drinking deterrent. Neices and nephews are also a great reproduction deterrent.
What does the attachment do?
https://www.navytimes.com/news/your-navy/2015/08/13/out-of-date-tiara-cloak-shed-from-navy-uniform-racks/
Oooooh
Once integrated biotronics becomes a thing, the implications of hacking will become a lot more serious.

Figure 24 : Structure d'un fichier de test csv.

2.2 Partie 2 : Prétraitement et Vectorisation

1. Prétraitement

L'objectif principal de cette étape est de réduire le texte, de le nettoyer à des mots simples, et de standardiser afin d'en faciliter l'utilisation. Avant de pouvoir entraîner nos modèles de classification, un prétraitement des données est nécessaire afin d'éviter de biaiser les performances des modèles.

Les étapes que nous suivons dans cette phase sont les suivantes :

- **Suppression des lignes vides.**
- **Suppression des linges dupliquées** car la répétition des lignes n'a pas d'importance, elle augmente seulement le volume des données.
- **Passage en minuscule :** Convertir toutes les lettres majuscules en minuscules.
- **Suppression des URL (liens).**
- **Suppression des nombres et des chiffres.**
- **Suppression des ponctuations :** Les ponctuations n'ont aucun effet sur l'analyse du texte, elles sont donc supprimées, car elles n'offrent aucune information utile pour la classification. Certains de ces ponctuations sont (, - ; ?. ! / @).
- **Remplacer les émojis par leurs noms** pour ne pas perdre leurs significations.
- **Suppression des mots courts :** Supprimer tous les mots de taille inférieure ou égale à 2.
- **Suppression des mots vides :** Les mots vides (ou stop words) sont des mots qui sont tellement communs qu'il est inutile de les traiter ou de les utiliser dans une recherche d'informations. En anglais certains de ces mots sont (a, about, you, it, him, …). Un mot vide est un mot non significatif figurant dans un texte, mais le cas de la négation est un cas important dans ce type de tâches qui peut changer complètement le sentiment d'une publication et comme le Corpus NLTK pour l'anglais contient 179 mots vides parmi eux nous trouvons certains termes de

négation (not, no, nor) nous proposons de créer notre corpus personnel qui ne contient pas les mots de négation.

- **La lemmatisation :** Est l'étape qui désigne l'analyse lexicale chargée de faire regrouper les mots d'une même famille qui partagent le même suffixe lexical. Chacun des mots du texte se trouve ainsi réduit en une entité appelée « Lemme ». Ce lemme désigne la forme canonique des mots. La lemmatisation regroupe les différentes formes que peut avoir un mot. Par exemple, un nom en pluriel va être réduit au singulier, un verbe à son infinitif, etc.

Par la suite nous proposons des exemples dans un tableau sur les étapes de prétraitement précédentes pour mieux comprendre comment les traiter.

Le paragraphe avant le prétraitement: Click here https://erisk.irlab.org .To Download the dataset of « 2020 »; To have access to the collections all participants have to fill, sign and send a user agreement?

Tableau 4 : Les résultats de prétraitement.

Prétraitement	**Paragraphe après le prétraitement**
1. Le passage en minuscule	click here https://erisk.irlab.org .to download the dataset of « 2020 » ; to have access to the collections all participants have to fill, sign and send a user agreement ?!
2. La suppression des URL	click here ~~https://erisk.irlab.org~~ .to download the dataset of « 2020 » ; to have access to the collections all participants have to fill, sign and send a user agreement ?!

3. La suppression des nombres et chiffres	click here .to download the dataset of « 2020 » ; to have access to the collections all participants have to fill, sign and send a user agreement ?!
4. La suppression des ponctuations	click here to download the dataset of « » ; to have access to the collections all participants have to fill sign and send a user agreement ?!
5. La suppression des mots courts	click here to download the dataset of to have access to the collections all participants have to fill sign and send a user agreement
6. La suppression des mots vides	click here download the dataset have access the collections all participants have fill sign and send user agreement
7. La lemmatisation	click download dataset have access **collection** all **participant** fill sign send user agreement

2. Vectorisation

L'objectif de cette étape est de transformer les mots en vecteurs de nombres réels pour les préparer à l'entrer dans l'apprentissage de nos modèles. Pour cela, nous passons par la tokenisation pour diviser les publications en mots puis nous utilisons le word embedding comme méthode de vectorisation et après nous faisons le padding pour fixer tous les publications à la même taille et à la fin nous utilisons le word2vec.

- **Tokenisation**

  La tokenisation est la division du texte brut en petits morceaux de mots ou de phrases, appelés jetons. Si le texte est divisé en mots, alors il est appelé « Tokenisation de mots » et s'il est divisé en phrases, il est appelé « Tokenisation de phrase ». Dans notre travaille nous utilisons la tokenisation par mot.

  Si nous prenons le paragraphe de la dernière étape de prétraitement (La lemmatisation), le résultat devient le suivant :

```
[ 'click' , 'download ' , 'dataset' , 'have' ,  'access' ,  'collection'  , 'all',
 'participant'  ,  'fill' ,'sign'  , 'send' ,  'user' , 'agreement' ]
```

Figure 25 : Le paragraphe après la tokenisation.

- **Padding (Rembourrage)**

  Elle consiste à créer des séquences de mots de la même taille en fixant une taille maximale, les publications de taille inférieure nous complétons leur remplissage avec des zéros, et pour les publications de taille supérieure nous ne prenons que les mots inférieurs à la taille maximale et ignorons le reste.

- **Word embedding**

  Le word embedding désigne un ensemble de techniques de l'apprentissage automatique qui vise à représenter les mots d'un texte par des vecteurs de nombres réels, projetés dans l'espace vectoriel [19].

  Cette représentation vectorielle améliore les performances des algorithmes d'apprentissage dans le domaine du traitement automatique de la langue.

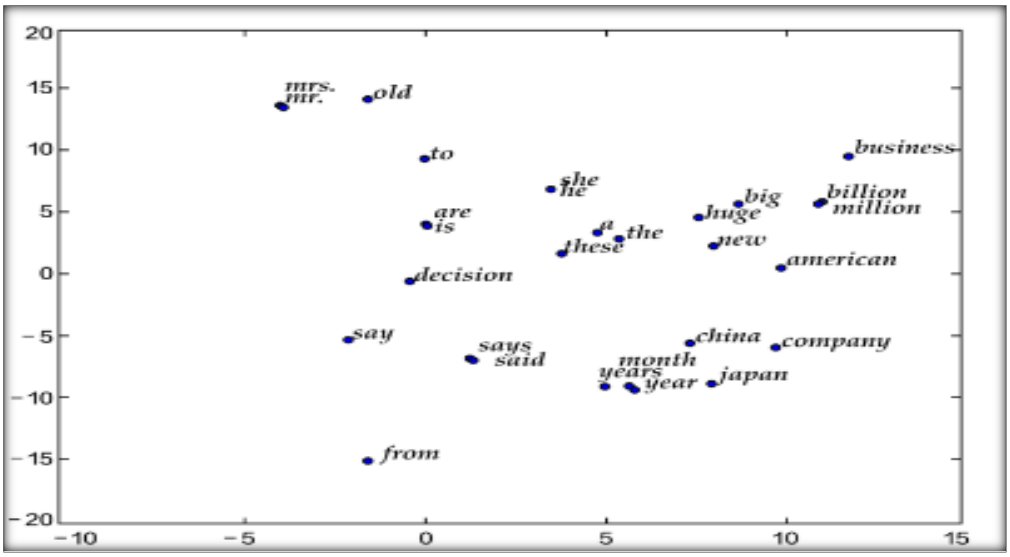


Figure 26 : Exemple d'une représentation vectorielle [61].

- **Word2Vec :**

C'est un algorithme de word embedding parmi les plus connus. Il repose sur des réseaux de neurones à deux couches et cherche à apprendre les représentations vectorielles des mots composant un texte, de telle sorte que les mots qui partagent des contextes similaires soient représentés par des vecteurs numériques proches [26] **(Figure 27)**.Ce dernier possède deux architecture neuronales appelées CBOW et Skip-Gram.

[26] https://dataanalyticspost.com/Lexique/word2vec/

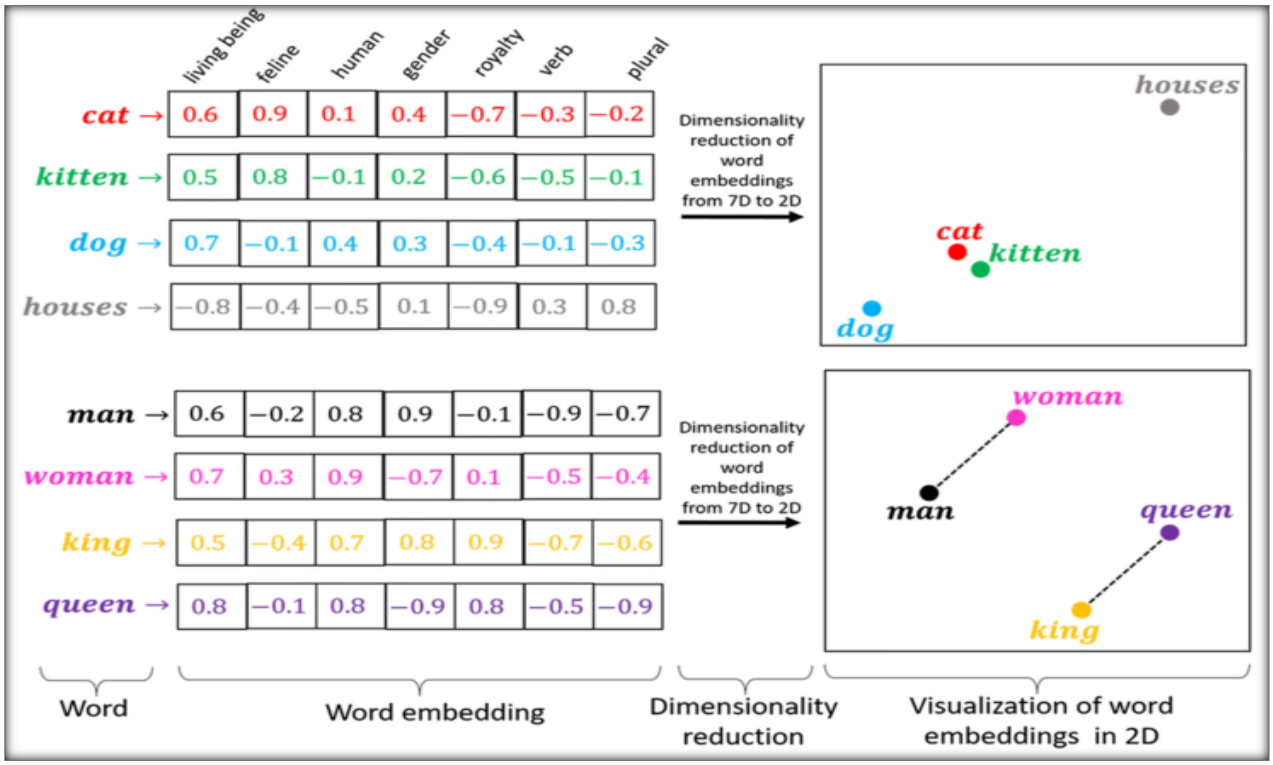


Figure 27 : Représentation de word2vec [61].

1. **<u>Modèle CBOW</u> :** « Continuous Bag of Words » Il entraîne le réseau de neurones pour prédire un mot en fonction de son contexte, le contexte correspond à un certain nombre de mots voisons à gauche et à droite du mot **(figure 28)**.

2. **<u>Modèle Skip-Gram :</u>** Le modèle skip-gram permet de prédire le contexte en fonction du mot. Il fonctionne bien avec une petite quantité de données d'entraînement **(figure 28)**.

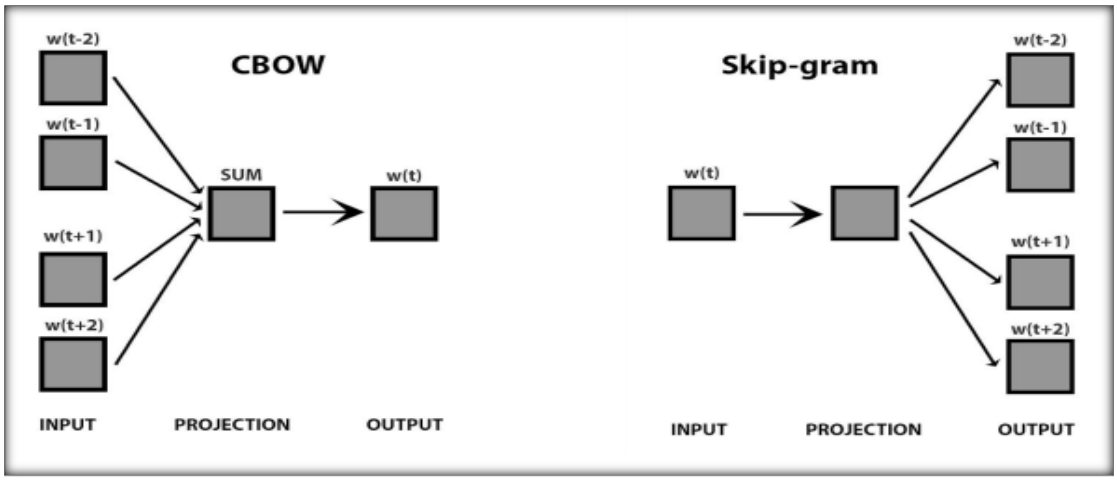


Figure 28 : Modèle Skip-gram et CBOW [62].

La **figure 29** illustre un exemple sur les deux modèles. Le côté gauche de la figure représente le modèle cbow qui prédit le mot cible, étant donné les mots sur ses côtés gauche et droit. En d'autres termes, il prédit «on», étant donné les mots «cat», «sat», «the» et «mat» dans son contexte. Le côté droit de la figure montre l'autre modèle qui est SkipGram où le contexte d'un mot est prédit étant donné le mot. Ici, le modèle apprend à prédire les mots contextuels «cat», «sat», «the» et «mat» étant donné le mot «on» [63].

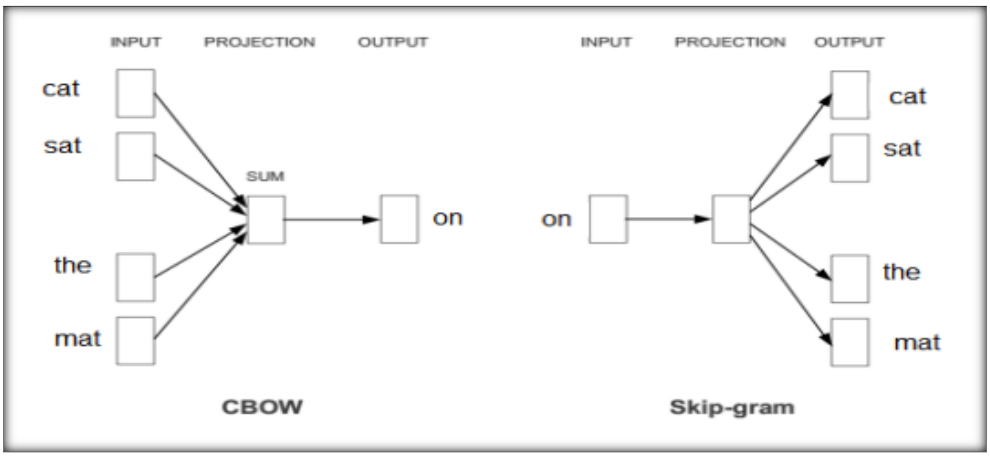


Figure 29 : illustre un exemple sur le modèle CBOW et SKIP-GRAM [63].

A la fin nous renforçons notre explication avec un algorithme qui résume la partie 2 de prétraitement des données et de véctorisation.

Algorithme Prétraitement_Vectorisation
Inputs : *Dataset*

Début
Lire*(Dataset) ;*
 # Etape de prétraitement

D1 = ***Remove_duplicate_lines*** *(Dataset) ;* // supprimer les linges dupliquées dans l'ensemble de données « Dataset »

D2 = ***Remove_empty_lines*** *(D1) ;* // supprimer les lignes vides dans l'ensemble de données « Dataset »

Text = D2 ['Text'] ; // obtenir tous les textes de la case de Text du Dataset

Tab = [ ] ; // créer un tableau vide

For i in Text :

T1 = ***Remove_numbers_and_digits*** *(i) ;* // Suppression des nombres et des chiffres

T2 = ***Remove_links*** *(T1) ;* // Supprimer tous les liens dans un texte

T3 = ***Set_to_lowercase*** *(T2) ;* // Mettre le texte en minuscule

T4 = ***Replace_emojis_with_their_names*** *(T3) ;* // Remplacer les émojis par leurs noms dans un texte

T5 = ***Remove_ponctuation*** *(T4) ;* // Supprimer la ponctuation

T6 = ***Remove_short_words*** *(T5) ;* // supprimer les mots courts

T7 = ***Remove_stopwords*** *(T6) ;* // Supprimer les mots vides « stopwords »

T8 = ***Lemmatization*** *(T7) ;* // Lemmatiser le texte

Tab = T8;

End For ;

 # Etape de véctorisation

toknizer = ***Tokenization*** *(Tab) ;* // deviser le tableau contenant les textes après le prétraitement par mot

sequences = ***Word_embedding*** *(Tab) ;* // *transformer les mots d'un texte par des vecteurs de nombres réels*

maxlength = 250 ; // identifier la longueur maximale des textes

X = ***Padding*** *(sequences, maxlength) ;* // créer des séquences de mots de la même taille en fixant une taille maximale « maxlength »

Word2vec = ***Word2vec*** *(toknizer) ;* // représenter les mots qui partagent des contextes similaires par des vecteurs numériques proches

End.

Figure 30 : Algorithme de prétraitement des données et de véctorisation.

2.3 Partie 3 : Les modèles d'apprentissage

Dans cette section, nous allons présenter nos modèles que nous avons utilisées pour calculer le degré de dépression. Nous avons créé deux modèles avec différentes architectures, en appliquant le premier modèle à CNN et le second modèle à l'Auto-encoder_CNN. Dans ce qui suit nous présentons l'architecture des deux modèles utilisées après le prétraitement et vectorisation :

1. Modèle CNN

Un CNN est un type de réseau neuronal artificiel utilisé dans la reconnaissance et le traitement d'images et spécifiquement conçu pour traiter les données de pixels. Récemment, il a été appliqué dans diverses tâches de TAL. Lorsque CNN est appliqué au texte au lieu à des images, il est nécessaire d'appliquer une représentation de tableau d'une dimension du texte, le texte est traité de la même manière que les images, à la fois pour l'extraction de caractéristiques et la classification. Le CNN utilise un système semblable à un perceptron multicouche qui a été conçu pour des besoins de traitement réduits.

La **figure 31** présente notre architecture de modèle CNN.

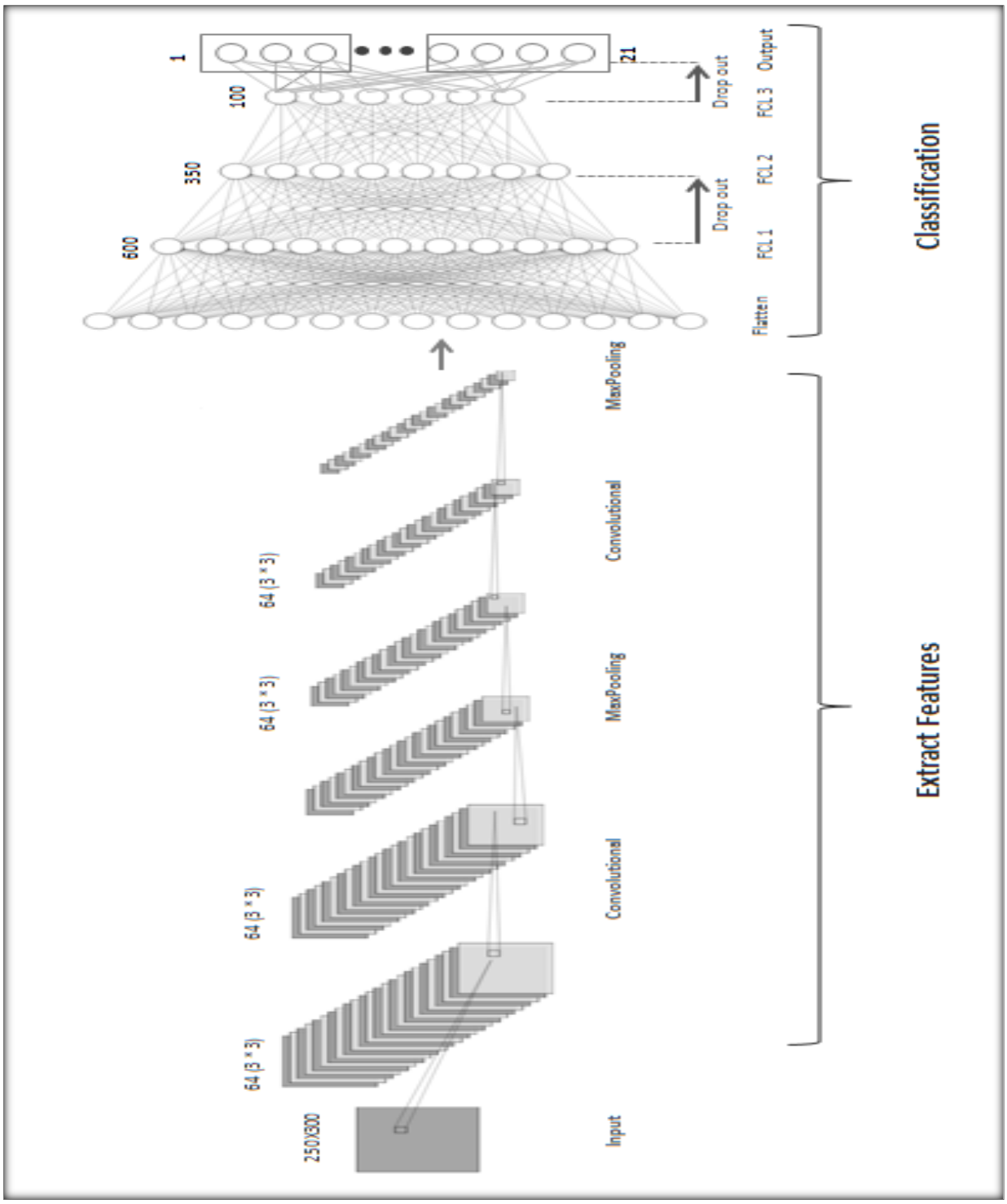


Figure 31 : L'architecture de modèle CNN.

Le premier modèle que nous présentons (CNN) est composé d'une couche d'embedding, quatre couches de convolution, deux couches de maxpooling, une couche de flatten (La couche d'entrée entièrement connectée), deux couches dropout, 3 couches entièrement connectée et 21 couches de sortie entièrement connectée.

- **Le texte en entrée** a une taille de 250 (pour 10942 publications), relié avec **une couche d'embedding** de dimension égale à 300 du modèle Word2vec Google, pour faire la combinaison entre le texte entré et la dernière étape de vectorisation qui est le word2vec. Pour cela nos entrées (des mots d'un texte représenté par un vecteur de nombres réels) deviennent représentées sous forme de matrice. Chaque ligne de la matrice correspond à un jeton, dans notre cas c'est un mot. Autrement dit, chaque ligne est un vecteur qui représente un mot. Par exemple pour une phrase de 7 mots utilisant un embedding de 300 dimensions, nous aurions une matrice 7 × 300 comme entrée (pour mieux comprendre voire la **figure 30** où un mot est représenté par un vecteur avec une dimension de 1 x 7 pour cela 7 est la dimension du modèle word2vec).
- **Les couches de convolution** composée de 64 filtres pour chaque couche, la taille de chaque filtre est de 3*3, La foulée du filtre est 1 pas, la fonction d'activation « RELU » est utilisée à chaque fois qu'on passe par une couche de convolution, cette fonction d'activation force les neurones à retourner des valeurs positives. Le but de ces couches est d'extraire des caractéristiques à partir de texte et nous utilisons ce nombre de couches car plus le nombre de couches est grand plus le modèle sera profond, mais aussi le nombre de couches que nous choisissons dépend des données et de leur taille et pour la dimension, nous choisissons des convolutions à une dimension, car nous traitons du texte.
- **La couche Maxpooling** est appliquée après chaque deux couches de convolution pour réduire la taille de texte et conserve les informations les plus essentielles.
- **La couche d'entrée entièrement connectée** (flatten) permet de convertir nos données de trois dimensions en vecteur d'une dimension car les couches entièrement connectées ne peuvent accepter que des données à une dimension.
- Dans **les couches entièrement connectées** nous avons utilisé 3 couches avec des neurones mentionnés successivement (600,350 ,100).

- **Les couches de dropout** appliqué après les couches entièrement connectées avec un pourcentage différent (30% et 50%), qui est une technique de régularisation pour réduire le sur-apprentissage dans les réseaux neurones.
- Dans **les couches de sortie entièrement connectée** nous avons utilisé **21 couches** avec des neurones différents selon les classes de réponse de chaque question du questionnaire BDI (3, 4, 4, …), chaque couche pour prédire une sortie par la fonction « SOFTMAX ». Nous utilisons la fonction « SOFTMAX » car nous traitons un problème de multi-class.

2. Modèle Auto-encoder_CNN

L'encodeur automatique(L'Auto-encoder) est un type de réseau de neurones qui utilisé pour apprendre une représentation compressée de données brutes.

Un Auto-encodeur est composé d'un encodeur et d'un décodeur. L'encodeur apprend à interpréter l'entrée et à la compresser pour créer une représentation compressée que nous appelons la couche latente et le décodeur tente de recréer l'entrée à partir de la version compressée fournie par l'encodeur(la couche latente). Après l'apprentissage, le modèle d'encodeur est enregistré et le décodeur est supprimé car notre but est de combiner la sortie de l'encodeur avec un autre modèle de classification qui est le cnn puisque l'autoencodeur seul ne peut pas faire la classification de text.

La **figure 32** présente notre architecture de modèle Auto-encoder_CNN.

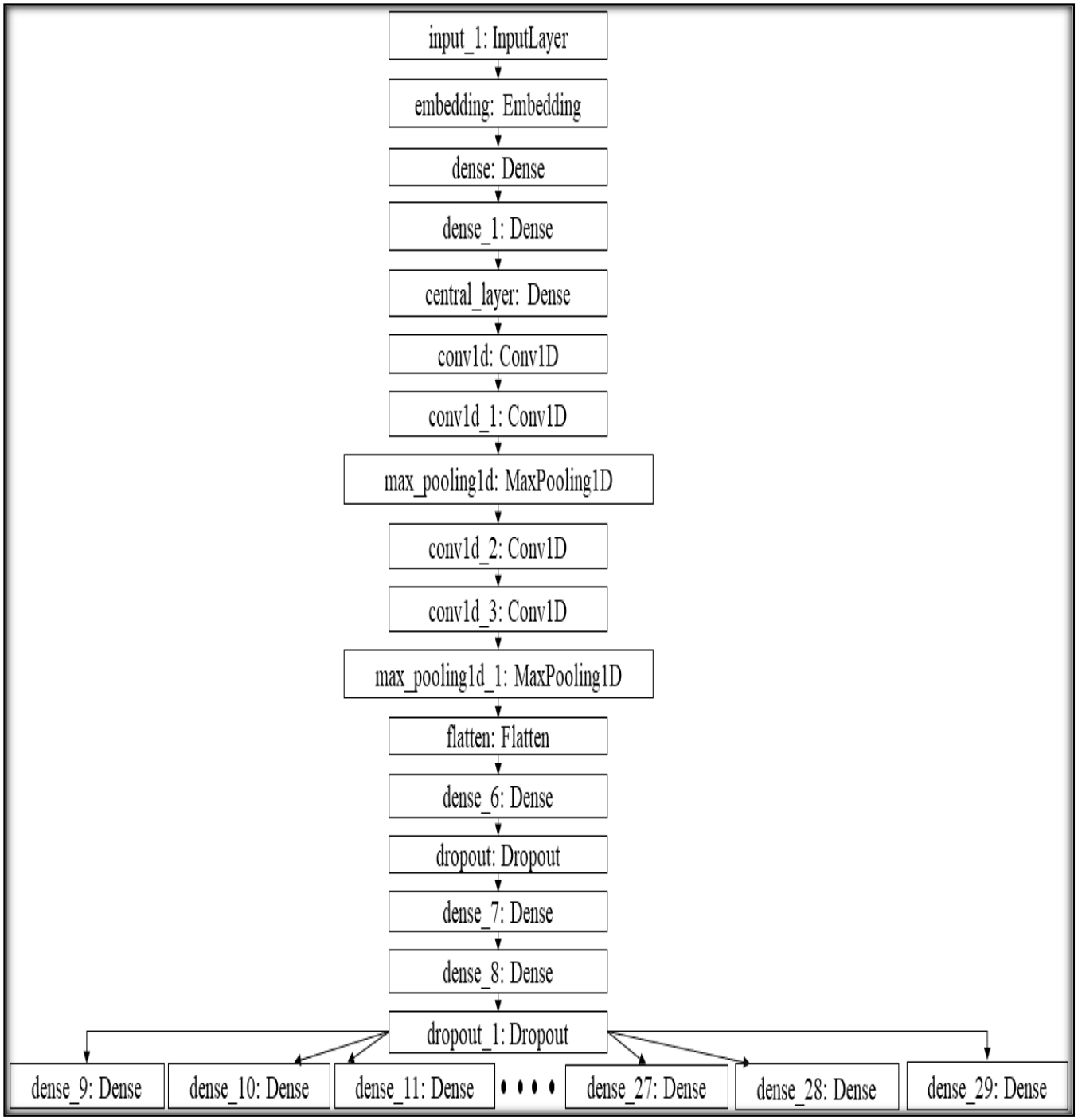


Figure 32 : L'architecture de modèle Auto-encoder_CNN.

Le deuxième modèle que nous présentons est une combinaison de modèle Auto-encoder avec l'architecture précédente de modèle CNN. Ce modèle est composé d'une couche embedding, deux couches encodeur, quatre couches de décodeur et une couche de latent représentation.

- **Le texte en entrée** a une taille de 250, relié avec **une couche d'embedding** de dimension égale à 300 du modèle Word2vec Google (nous avons expliqué cette étape plus dans l'architecture de modèle CNN).
- **La couche embedding** est reliée avec **les couches encodeur** avec des neurones mentionnés successivement (200 ,100) et une fonction d'activation « RELU ».
- **La couche latent representation** a 30 neurones et une fonction d'activation « RELU ». Cette couche est l'entrée de **la première couche de convolution** du modèle CNN, dans cette partie la combinaison est faite entre les deux modèles.

3.3 Partie 4 : Prédiction des résultats

Dans cette dernière partie nous faisons la prédiction des réponses pour les 21 questions de questioannaire BDI. Cette prédiction est faite par la fonction d'activation « SOFTMAX » qui attribue trois ou quatre probabilités décimales pour chaque question selon le nombre de réponses de la question prédéfinit dans le questionnaire BDI (par exemple la question 3 a quatre réponses possibles sont 0, 1, 2, 3), dont la valeur maximale est la prédiction finale de la réponse de la publication.

Après prédire les réponses d'une questionne pour toutes les publications d'un Subject et tandis qu'il faut uniquement 21 réponses pour chaque sujet (c'est-à-dire une réponse pour chaque question), nous utilisons une méthode statistique pour générer 1 exécution (run) pour chaque modèle. La méthode statistique consiste à calculer la fréquence de chaque réponse générée à une question, dont la plus fréquente est celle qui prend comme réponse finale à la question.

Par exemple dans la première ligne de la **figure 33** nous présentons la prédiction de toutes les publications d'un sujet et par la suite nous calculons la fréquence des réponses (la réponse 0 apparait 43 fois, la réponse 1 apparait 64 fois, la réponse 2 apparait 12 fois, la réponse

3 apparait 0 fois) et à la fin nous prenons la réponse 1 comme réponse finale à la question car c'est la plus fréquentes.

```
argmax [0, 0, 1, 2, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 2, 0, 0, 0, 1, 0, 0, 2, 1, 0, 1, 0, 1, 0, 2, 0, 1, 0, 1, 0, 2, 1, 1, 0, 0, 1
argm_val0 43
argm_val1 64
argm_val2 12
argm_val3 0
64
argmax [0, 0, 1, 3, 1, 1, 0, 1, 0, 3, 1, 3, 3, 3, 3, 0, 0, 0, 0, 3, 2, 0, 0, 0, 1, 0, 0, 2, 3, 0, 0, 0, 1, 0, 2, 0, 1, 0, 3, 0, 2, 0, 1, 0, 1, 1
argm_val0 41
argm_val1 42
argm_val2 10
argm_val3 26
42
argmax [0, 0, 1, 3, 0, 2, 0, 1, 0, 3, 2, 3, 3, 0, 3, 2, 0, 0, 2, 3, 1, 2, 2, 2, 2, 0, 0, 1, 0, 0, 1, 0, 2, 0, 1, 2, 2, 0, 3, 2, 1, 1, 0, 1, 1, 2
argm_val0 47
argm_val1 24
argm_val2 31
argm_val3 17
47
```

Figure 33 : exemple de prédiction pour chaque question d'un sujet.

Dans la **figure 34** nous avons schématisé le fonctionnement de cette partie.

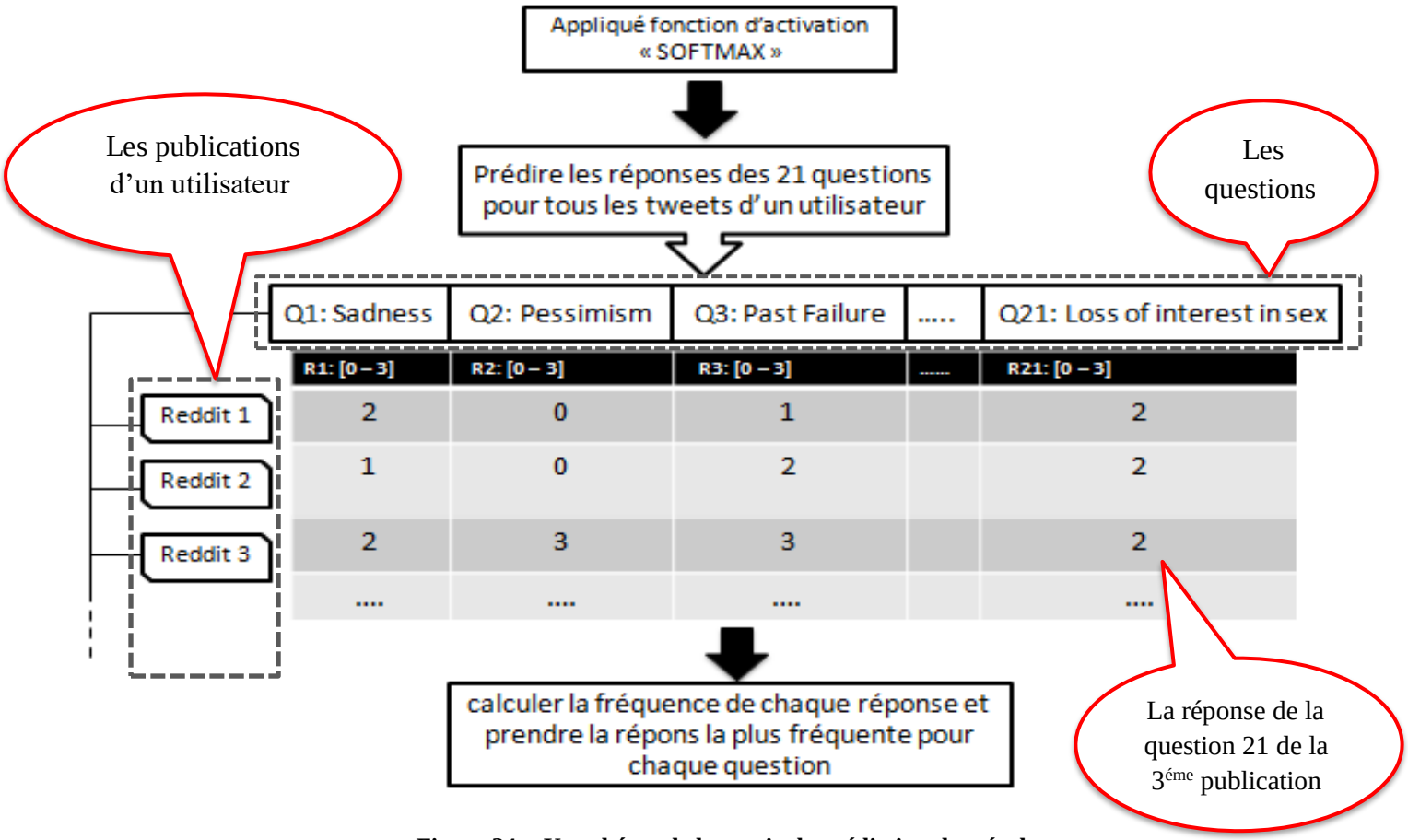
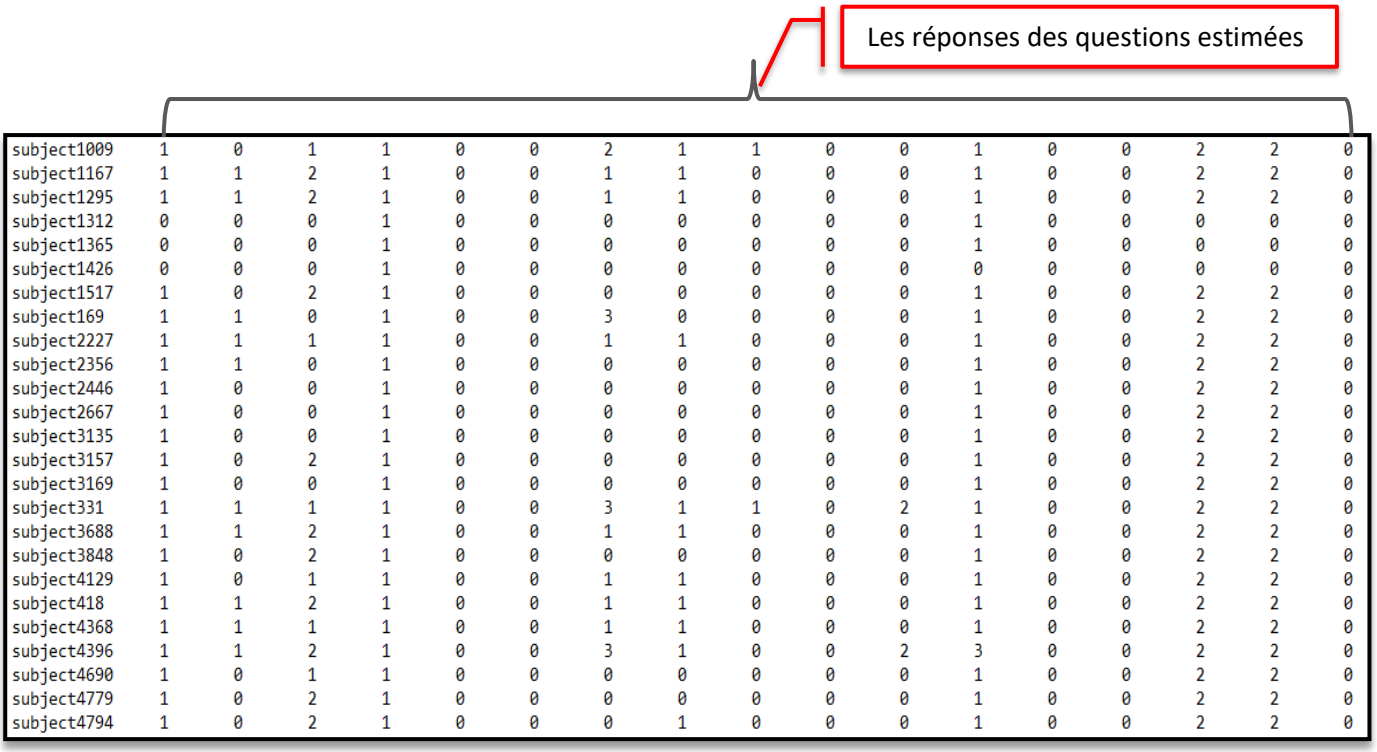


Figure 34 : Un schéma de la partie de prédiction des résultats.

A la fin nous envoyons deux fichiers TXT contenant les résultats obtenus à partir de chaque modèle à eRisk pour les évaluer (comme le montre la **figure 35**).



subject1009	1	0	1	1	0	0	2	1	1	0	0	1	0	0	2	2	0
subject1167	1	1	2	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject1295	1	1	2	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject1312	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
subject1365	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
subject1426	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
subject1517	1	0	2	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject169	1	1	0	1	0	0	3	0	0	0	0	1	0	0	2	2	0
subject2227	1	1	1	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject2356	1	1	0	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject2446	1	0	0	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject2667	1	0	0	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject3135	1	0	0	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject3157	1	0	2	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject3169	1	0	0	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject331	1	1	1	1	0	0	3	1	1	0	2	1	0	0	2	2	0
subject3688	1	1	2	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject3848	1	0	2	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject4129	1	0	1	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject418	1	1	2	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject4368	1	1	1	1	0	0	1	1	0	0	0	1	0	0	2	2	0
subject4396	1	1	2	1	0	0	3	1	0	0	2	3	0	0	2	2	0
subject4690	1	0	1	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject4779	1	0	2	1	0	0	0	0	0	0	0	1	0	0	2	2	0
subject4794	1	0	2	1	0	0	0	1	0	0	0	1	0	0	2	2	0

Figure 35 : Partie de fichier TXT du modèle CNN.

3 Conclusion

Ce chapitre montre toutes les étapes de conception avec des explications pour rapprocher la compréhension. Nous avons choisi deux modèles différents, plus connu par leurs performances dans le domaine DL pour obtenir des meilleurs résultats. Le prochain chapitre mettra en évidence la partie expérimentale de notre projet où nous nous concentrerons sur les outils et bibliothèques utilisés pour implémenter et réaliser notre système. A la fin, une représentation et discussion sur les résultats obtenus seront aussi étalés.

Chapitre 04 : Expérimentations et résultats

1 Introduction

L'implémentation est la phase la plus importante après celle de la conception. Le choix des outils de développement influence énormément sur le coût en temps de programmation, ainsi que sur la flexibilité du produit à réaliser. Cette phase consiste à transformer le modèle conceptuel établi précédemment en des composants logiciels formant notre système.

Dans les chapitres précédents, nous avons détaillé la méthodologie suivie pour notre travail ainsi que la définition de notre approche. Dans ce chapitre, nous procéderons à la présentation de l'environnement de travail, et nous conclurons enfin par les principaux résultats obtenus selon nos différentes approches mises en place, à travers des digrammes représentatifs.

2 Environnement de travail

L'environnement de travail est constitué par deux parties nommées environnement matériel et environnement logiciel.

2.1 Environnement matériel

Nous avons utilisé deux machines différentes. La première est dotée d'un Intel(R) Core (™) i5-2520M CPU @ 2.50GHz processeur x64 avec 4.00Go RAM et la deuxième possède un Intel(R) Pentium(R) CPU B960 @ 2.20GHz processeur x64 avec 4.00Go RAM.

2.2 Environnement logiciel

Le choix de l'environnement de programmation convenable est très important pour le développement des projets.

Le langage que nous avons adopté est Python 3.8 sous l'environnement de Google Colaboratory (Colab), notre choix c'est porté sur ce langage à cause de sa simplicité ; la syntaxe des lignes de code présente une certaine clarté, ce qui en facilite la lecture et la compréhension, permet aux développeurs de créer des fonctions avec moins de lignes de code, ce qui ne serait pas le cas avec d'autres langages de programmation, ses bibliothèques diverses et riches, celles dédiées au l'apprentissage en profondeur et celles utilisées pour la gestion d'autres structures de données et la dernière cause est le code Python peut s'exécuter sur n'importe quelle machine, que ce soit Linux, Mac ou Windows.

Google Colaboratory : Google Colaboratory est un service dans le cloud, proposé par Google gratuitement. Il est basé sur l'environnement Jupyter Notebook et est destiné à la formation et à la recherche en apprentissage automatique. Google Colab est un outil utilisé pour former et tester rapidement différents modèles d'apprentissage automatique sans restriction matérielles. Sa particularité est que tout le monde peut l'utiliser, il ne nécessite aucune configuration [27].

Les principales fonctionnalités de Google Colab sont:

- Aucune de configuration.

- Offre des bibliothèques python.

- Accès gratuitement au puissant matériel Google (GPU et TPU) sans affecter les performances de votre PC.

- La possibilité d'enregistrer et de partager tous les blocs-notes développer ou exécuter dans Colab.

3 Librairies

Nous citons ci-dessous les librairies utilisées pour modéliser notre projet :

Pandas: Pandas est une librairie Python qui nous a permis de manipuler facilement nos données pour l'analyse et le prétraitement [64].

NumPy: La bibliothèque NumPy permet d'effectuer des calculs numériques avec Python [64].

Keras : Keras est une bibliothèque Python open source gratuite puissante et facile à utiliser pour le développement et l'évaluation de modèles d'apprentissage en profondeur.

Il englobe les bibliothèques de calcul numérique efficaces Theano et TensorFlow et il permet de définir et de former des modèles de réseaux neuronaux en seulement quelques lignes de code [28].

[27] https://www.informatique-mania.com/linformatique/google-colaboratory/

[28] https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/

NLTK : (Natural Language Toolkit)est une bibliothèque logicielle en Python permettant un traitement automatique des langues, développée par Steven Bird et Edward Loper du département d'informatique de l'université de Pennsylvanie. En plus de la bibliothèque, NLTK fournit des démonstrations graphiques, des données-échantillon, des tutoriels, ainsi que la documentation de l'interface de programmation (API) [29].

Tensorflow : Est une bibliothèque Python pour le calcul numérique rapide créée et publiée par Google. Il s'agit d'une bibliothèque de base qui peut être utilisée pour créer des modèles de l'apprentissage en profondeur directement ou en utilisant des bibliothèques d'en capsuleurs qui simplifient le processus construit au-dessus de TensorFlow [30].

Gensim : Est une bibliothèque Python pour la modélisation de sujets, l'indexation de documents et la recherche de similitudes avec de grands corpus. Le public cible est la communauté du traitement du langage naturel (PNL) et de la recherche d'informations (IR) [31].

Scikit-learn : (sklearn) Est probablement la bibliothèque la plus utile pour l'apprentissage automatique en Python. La bibliothèque sklearn contient de nombreux outils efficaces pour l'apprentissage automatique et la modélisation statistique, notamment la classification, la régression, le regroupement et la réduction de la dimensionnalité [32].

Regular expression : 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) [33].

[29] https://fr.wikipedia.org/wiki/Natural_Language_Toolkit
[30] https://machinelearningmastery.com/introduction-python-deep-learning-library-tensorflow/
[31] https://pypi.org/project/gensim/
[32] https://www.analyticsvidhya.com/blog/2015/01/scikit-learn-python-machine-learning-tool/

[33] https://docs.python.org/3/library/re.html

4 DataSet

Dans cette section, nous expliquerons en détail la structure d'ensemble de données (dataset) utilisée dans la mise en œuvre et la prédiction de notre projet.

L'ensemble de données se compose de publications Reddit et de commentaires de personnes dépressives, stockés dans des fichiers XML nommé par « Subject » pour chaque personne (**figure 36**).

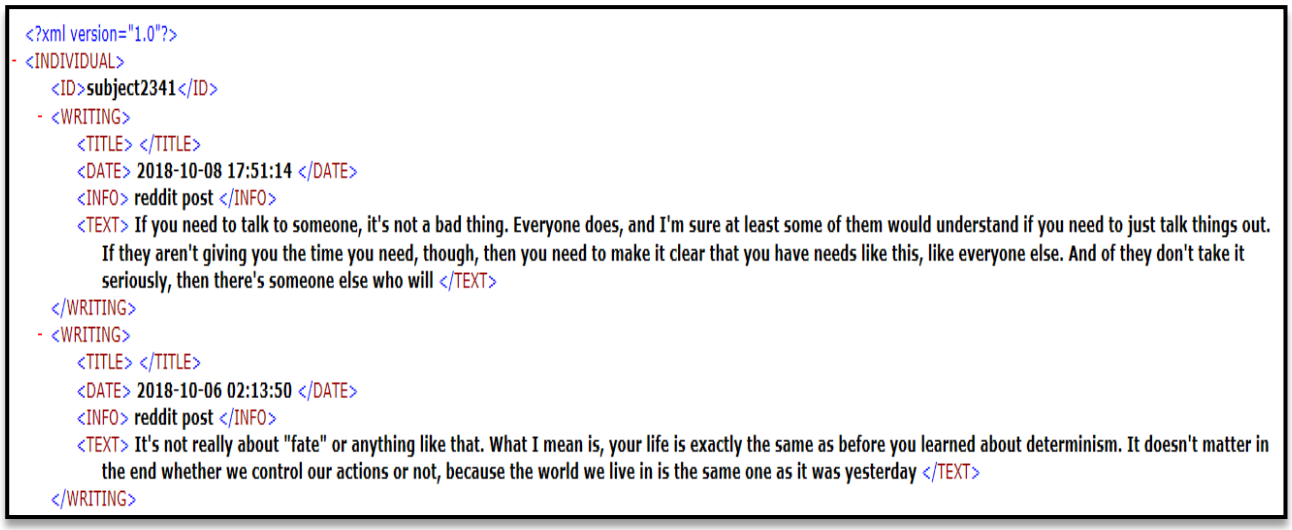


Figure 36 : Partie d'un fichier XML de subject2341.

Ainsi, l'ensemble de données se compose de questionnaires sont dérivés du BDI [Annexe n°1] dans un fichier TXT , ces questionnaires évalue la présence de sentiments comme la tristesse, le pessimisme, la perte d'énergie, etc. ont été remplis par les utilisateurs des médias sociaux avec l'historique des écrits de chaque utilisateur, ces questionnaires remplis sont les données de vérité de base appelée en anglais « ground truth data » qui serviront à l'évaluation des performances des systèmes des participants au laboratoire.

Chaque ligne du fichier TXT a un identifiant d'utilisateur par exemple (subject436) et 21 valeurs. Ces valeurs correspondent aux réponses aux questions du questionnaire de BDI (les valeurs possibles sont 0, 1a, 1b, 2a, 2b, 3a, 3b pour les questions « 16 et 18 » et 0, 1, 2, 3 pour le reste des questions) (**Tableau 5**).

Tableau 5 : Structure du fichier des réponses du questionnaire de "BDI".

subject2341	1	2	3	2	3	2	3	3	1	0	0	2	3	3	2	2b	2	3b	2
subject2827	1	3	3	2	3	2	2	3	2	2	1	1	2	3	1	2b	1	0	2
subject9218	2	2	1	2	1	0	3	2	1	2	2	3	1	2	2	2a	0	2a	2
subject9454	1	3	3	2	1	3	3	3	2	3	2	3	3	2	2	3b	1	3a	3
subject1272	1	2	2	1	0	1	3	2	1	3	3	1	2	2	2	3b	0	3a	1
subject2961	1	1	1	1	0	0	1	1	1	1	1	1	1	0	2	1b	0	3b	1

Afin d'entrainer nos modèles nous avons utilisé l'ensemble de données 2019 qui a été fourni par eRisk qui contient 4 sujets « Subject » de la catégorie « **dépression minimale** (minimal depression) », 4 sujets « Subject » de la catégorie « **dépression légère** (mild depression) », 4 sujets « Subject » de la catégorie « **dépression modérée** (moderate depression) », et enfin 8 sujets « Subject » de la catégorie « **dépression sévère** (severe depression) » ce qui fait un total de 20 sujets d'étude (**figure 37**) avec 10941 publications.

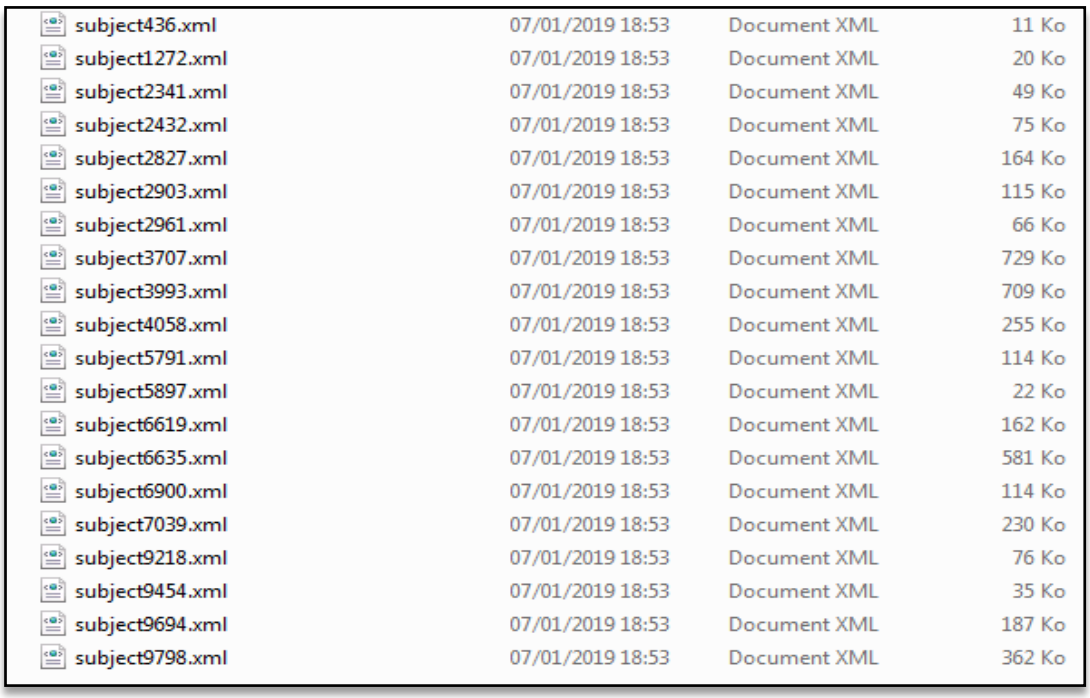


Figure 37 : Liste des subjects.

Pour tester nos modèles nous avons utilisé l'ensemble de données 2020 qui a été fourni par eRisk qui contient 10 sujets « Subject » de la catégorie « **dépression minimale** (minimal depression) », 23 sujets « Subject » de la catégorie « **dépression légère** (mild depression) », 18 sujets « Subject » de la catégorie « **dépression modérée** (moderate depression) », et enfin 19 sujets « Subject » de la catégorie « **dépression sévère** (severe depression) » ce qui fait un total de 70 sujets d'étude avec 35562 publications.

5 Mesure de performance

Dans le domaine psychologique, il est courant d'associé les niveaux de dépression aux catégories suivantes:

dépression minimale (niveaux de dépression 0-9)

dépression légère (niveaux de dépression 10-18)

dépression modérée (niveaux de dépression 19-29)

dépression sévère (niveaux de dépression 30-63)

- **Taux de réussite « Hit Rate (HR) » :** Est une mesure qui calcule le ratio de cas où le questionnaire automatique a exactement la même réponse que le questionnaire réel. Par exemple, une prédiction où 14 des 21 questions du BDI correct ont obtiennent une valeur **HR** de 14/21.
- **Taux de réussite moyen « Average Hit Rate (AHR) » :** Est le Taux de réussite (HR) pour tous les utilisateurs.
- **Taux de proximité « Closeness Rate CR »:** Le CR tient compte du fait que les réponses au questionnaire du BDI représentent une échelle de classement.

Pour chaque question, CR calcule la différence absolue (da) entre la réponse réelle et la réponse automatique, cette différence absolue est transformée en un score d'efficacité comme suit: CR = (mda - da) / mda, où mda est la différence absolue maximale.

Cependant, les questions #16 et #18 ont sept réponses possibles (0; 1a; 1b; 2a; 2b; 3a; 3b), où les paires (1a; 1b), (2a; 2b), (3a; 3b) sont considérés comme équivalents car ils reflètent le même niveau de dépression. En conséquence, la différence entre 3b et 0 est égale à 3 (et la différence entre 1a et 1b est égale à 0).

- **Taux de proximité moyen « Average Closeness Rate (ACR) »:** Est le taux de proximité (CR) en moyenne pour tous les utilisateurs.
- **Différence entre les niveaux globaux de dépression « Difference between**

overall depression levels (DODL) » : Cette mesure calcule le niveau de dépression global (somme de toutes les réponses) pour le questionnaire réel et automatique. De plus, la différence absolue (da globale) entre le score de dépression réel et le score de dépression prévu est calculée. Les niveaux de dépression sont des entiers compris entre 0 et 63, ces nombres sont dérivés de l'addition des nombres de réponses du BDI. La mesure DODL est normalisé dans l'intervalle [0,1] comme suit: DODL = (63 - da global) / 63.

- **Différence entre les niveaux globaux de dépression moyen « Average DODL (ADODL) » :** Est la différence entre les niveaux globaux de dépression (DODL) en moyenne pour tous les utilisateurs.
- **Taux de succès de la catégorie de dépression « Depression Category Hit Rate (DCHR) »:** La dernière mesure d'efficacité consiste à calculer la fraction de cas où le questionnaire prévue conduit à une catégorie de dépression équivalente à la catégorie de dépression obtenue à partir du questionnaire réel.

6 Résultats et évaluations des performances

Dans la dernière partie de notre recherche, nous présenterons nos résultats après l'évaluation et les comparerons avec les résultats des exécutions de l'année dernière et ceux soumis pour la tâche à la conférence CLEF eRisk2020.

L'année dernière, Au total 5 équipes ont soumis 17 essais différents pour la tâche 2 d'eRisk2020.

Les diagrammes ci-dessous montrent nos résultats par rapport à l'équipe USDB de l'année dernière [56] et les meilleurs résultats obtenuent l'année passée pour chaque métrique.

En comparant seulement les exécutions que nous avons faites, la première exécution utilisant CNN a obtenu les meilleurs résultats en AHR avec 37,96% des réponses correctes, en ACR avec 69,32% , en ADODL avec 79,02% et en DCHR avec 27,14%.

Les 2 exécutions que nous avons effectuées ont obtenu les meilleurs résultats en AHR et ACR par rapport les résultats de l'année dérniere d'équipe USDB [56] et des bonnes résultats par rapport les meilleurs résultats de l'année passée. Et des scores proches pour l'ADODL, concernent le DCHR l'exécution 1 a obtenu les meilleurs résultats.

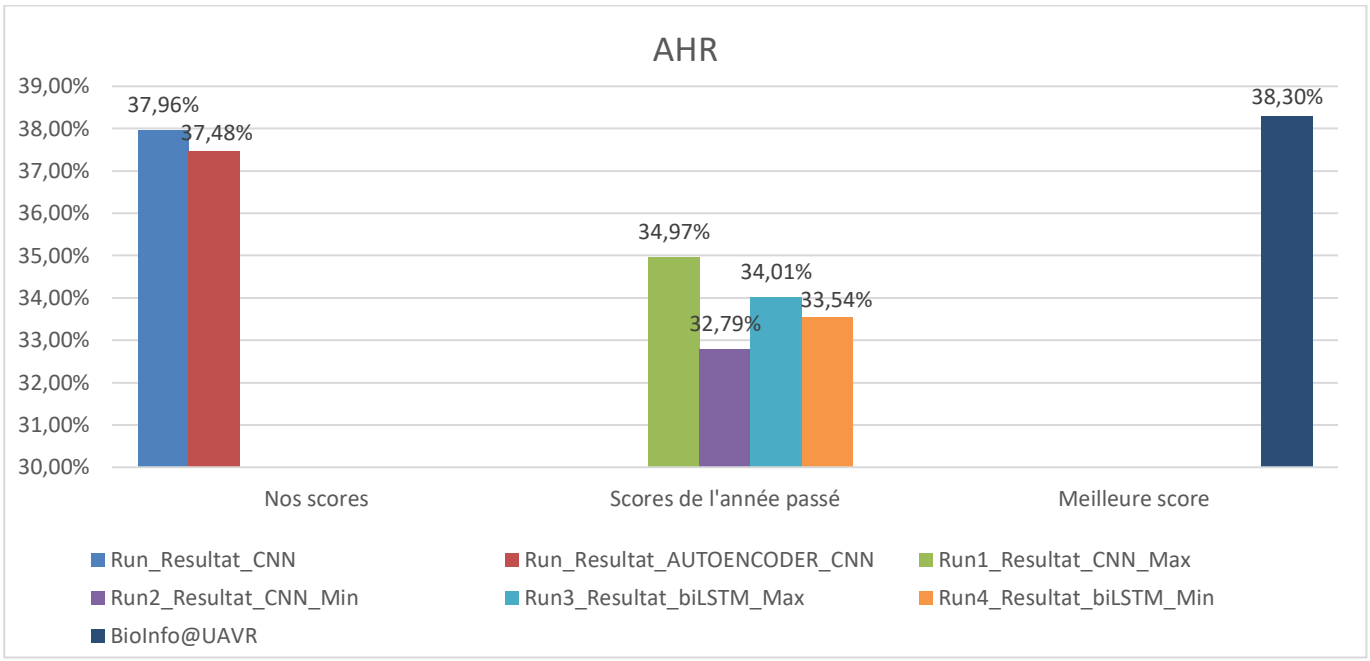


Figure 38 : Diagramme de comparaison de nos résultats avec les résultats de l'année passée d'équipe USDB et les meilleurs résultats obtenus pour la mesure AHR au challenge eRisk2020.

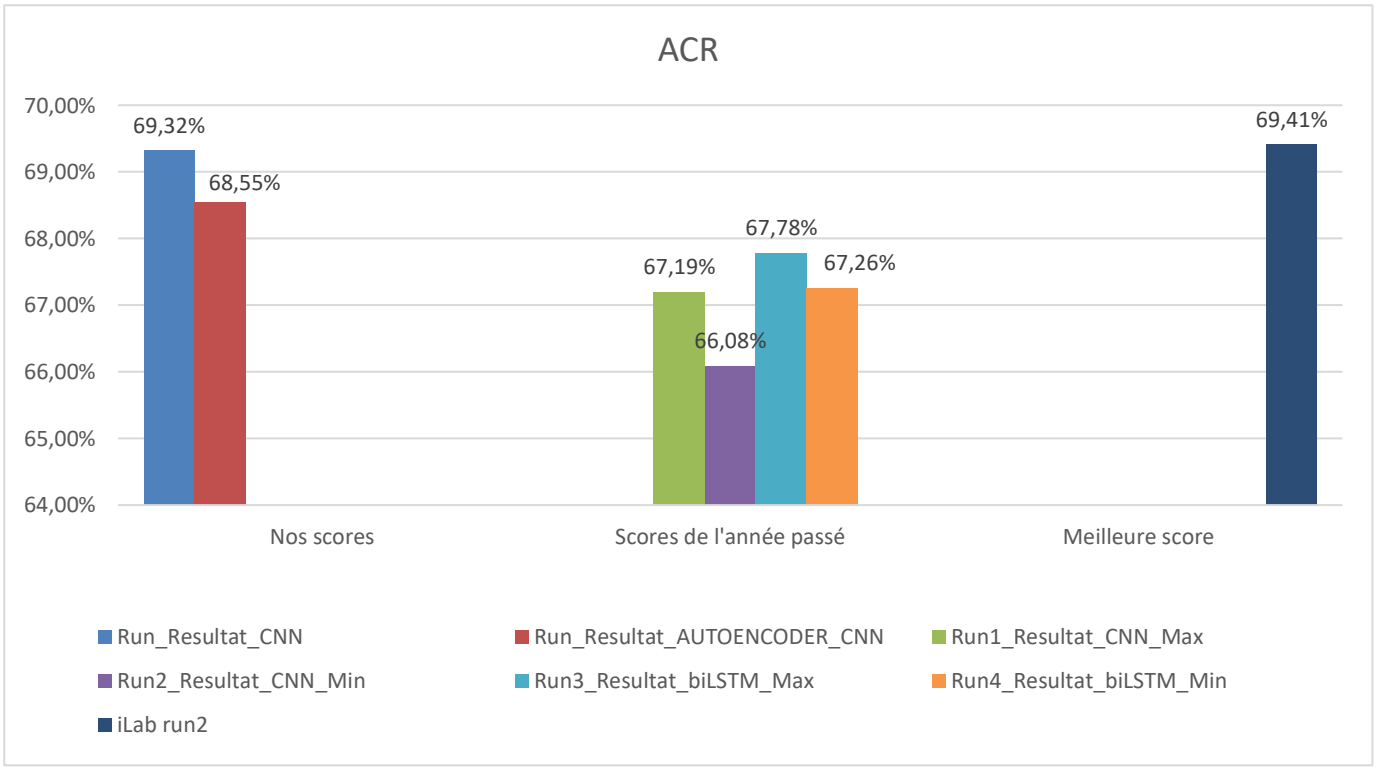


Figure 39 : Diagramme de comparaison de nos résultats avec les résultats de l'année passée d'équipe USDB et les meilleurs résultats obtenus pour la mesure ACR au challenge eRisk2020.

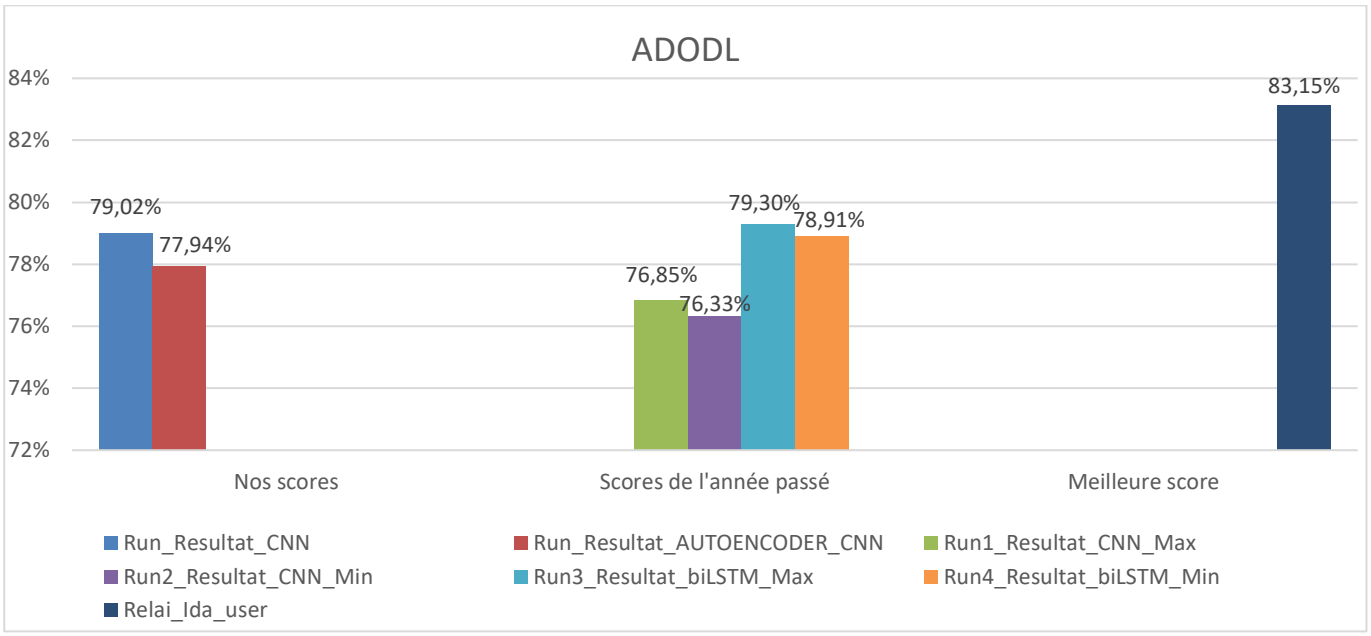


Figure 40 : Diagramme de comparaison de nos résultats avec les résultats de l'année passée d'équipe USDB et les meilleurs résultats obtenus pour la mesure ADODL au challenge eRisk2020.

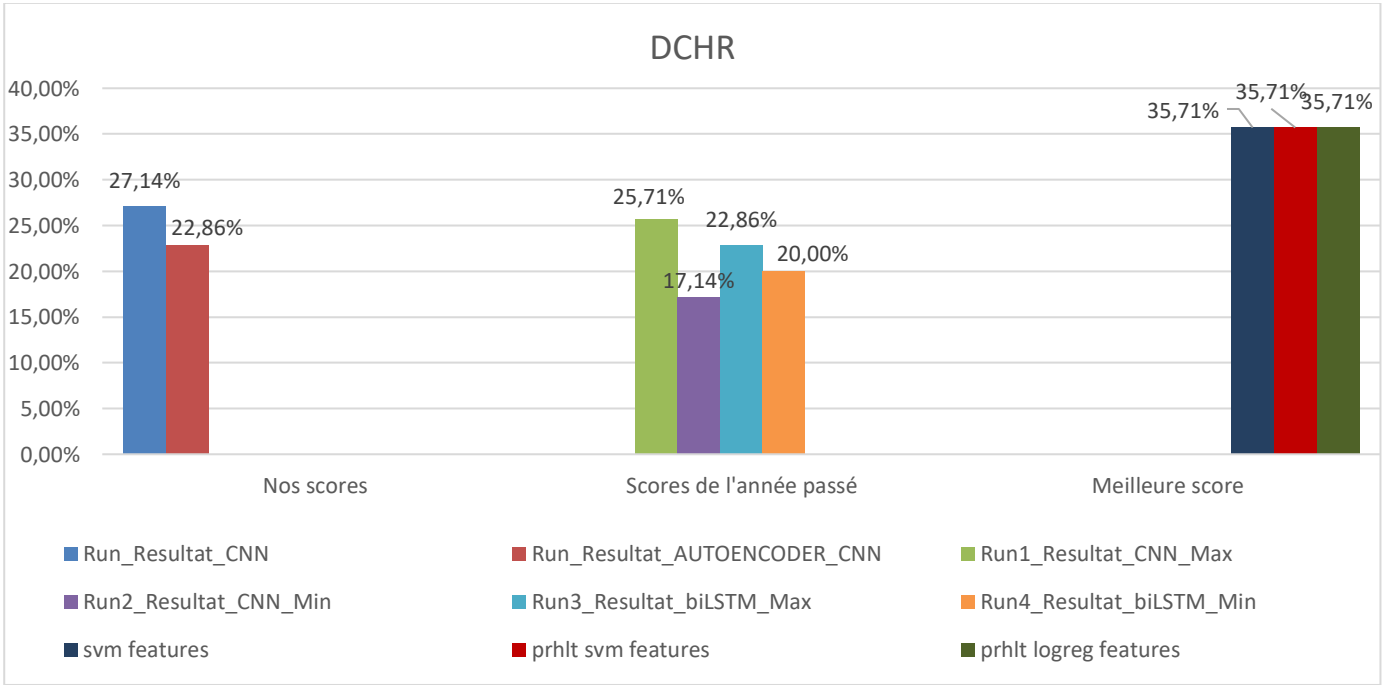


Figure 41 : Diagramme de comparaison de nos résultats avec les résultats de l'année passée d'équipe USDB et les meilleurs résultats obtenus pour la mesure DCHR au challenge eRisk2020.

Suite à la comparaison et aux diagrammes, nous avons conclu que le réseau de neurones convolutifs dans le modèle que nous avons proposé pour calculer le degré de dépression donne des résultats légèrement meilleurs par rapport à Auto-encoder_CNN. De sorte que les valeurs d'AHR et ACR sont très proches, et pour d'ADODL et DCHR sont légèrement écartées.

Nous remarquons également que l'exécution de notre modèle CNN a obtenu les meilleurs résultats en AHR, ACR, ADODL et DCHR par rapport le modèle CNN de l'année dernière [56]. Le tableau ci-dessous représente la comparaison entre le modèle CNN de l'année derniere de l'équipe USDB [56] et notre modèle.

Tableau 6 : Comparaison entre les résultats de notre modèle CNN et de l'USDB.

Résultats	AHR	ACR	ADODL	DCHR
Résultat de modèle CNN de l'année dernière	34.97%	67.19%	76.85%	25.71%
Résultat de notre modèle CNN	**37.96%**	**69.32%**	**79.02%**	**27.14%**

Répondre au questionnaire BDI avec un ensemble de données de 20 sujets d'entrainement seulement avec 10942 publication s'est avéré être tout un défi. Le tableau ci-dessous résume les résultats de tous les participants à la tâche 2 d'eRisk2020 et nos résultats. En général, les résultats de cette tâche sont assez cohérents avec peu de variation d'une équipe à l'autre. Cela peut être considéré comme une indication de la difficulté de la tâche.

Tableau 7 : La performance des résultats eRisk2020 et nos résultats.

Executions	AHR	ACR	ADODL	DCHR
BioInfo@UAVR	**38.30%**	69.21%	76.01%	30.00%
iLab run1	36.73%	68.68%	81.07%	27.14%
iLab run2	37.07%	**69.41%**	81.70%	27.14%

iLab run3	35.99%	69.14%	82.93%	34.29%
prhlt logreg features	34.01%	67.07%	80.05%	**35.71%**
prhlt svm use	36.94%	69.02%	81.72%	31.43%
prhlt svm features	34.56%	67.44%	80.63%	**35.71%**
svm features	34.56%	67.44%	80.63%	**35.71%**
relai context paral user	36.80%	68.37%	80.84%	22.86%
relai context sim answer	21.16%	55.40%	73.76%	27.14%
relai Ida answer	28.50%	60.79%	79.07%	30.00%
relai Ida user	36.39%	68.32%	**83.15%**	34.29%
relai sylo user	37.28%	68.37%	80.70%	20.00%
Run1 resultat CNN Methode max	34.97%	67.19%	76.85%	25.71%
Run2 resultat CNN Methode suite	32.79%	66.08%	76.33%	17.14%
Run3 resultat BILSTM Methode max	34.01%	67.78%	79.30%	22.86%
Run4 resultat BILSTM Methode suite	33.54%	67.26%	78.91%	20.00%
Run_Resultat_CNN	**37.96%**	**69.32%**	79.02%	27.14%
Run_Resultat_Auto-encoder_CNN	**37.48%**	68.55%	77.94%	22.86%

Avec le modèle CNN, nous avons obtenu la deuxième place dans les métrique de l'AHR avec un score de 37.96% et de l'ACR avec un score de 69.32% et des scores proches pour le reste avec un score de 79.02% d'ADODL et 27.14% de DCHR.

Avec le modéle Auto-encoder_CNN, nous avons obtenu la deuxième place dans la métrique AHR avec un score de 37.48% et des résultats proche pour le reste avec un score de 68.55% d'ACR , 77.94% d'ADODL et 22.86% de DCHR.

Bien que nous n'ayons effectué que deux exécution, nous avons obtenu des bons résultats par rapport à ceux qui ont effectué quatre et sinq exécutions.

Nous conclurons que le modèle CNN a donné des résultats de performance très favorables pour toutes les mesures par rapport l'année derniere, et excellents surpassant tous les modèles qui ont utilisé par l'équipe USDB [56] à la conférence CLEF eRisk2020. Cela nous amène à croire que si nous augmentons l'entrainement du modèle par plus de 100 epochs, cela améliorera ses performances pour prédire des résultats étonnants .

La combinison de l'Auto-encoder et du modèle CNN a également donné des meilleurs résultats en AHR et ACR, et des résultats très proches pour les autres mesures par rapport l'année derniere. Cela nous amène à croire que si nous modifions l'architecture du modèle Auto-encoder_CNN, cela améliorera ses performances pour prédire des meilleurs résultats.

7 Conclusion

Dans ce chapitre, nous avons mentionné les principaux outils et packages qui ont été mis en œuvre dans notre programme, afin d'atteindre notre objectif de prédire le degré de dépression chez une personne. Et nous avons également présenté toutes les parties que nous avons effectué pour obtenir les résultats que nous avons montrés pour deux modèles différents. Ci-dessous, nous présentons nos conclusions et points de vue généraux.

Conclusion Générale

L'objectif de notre étude était de résoudre l'un des plus grands problèmes de santé mentale, qui est la détection précoce de la dépression. Plus spécifiquement, nous avons choisi de nous concentrer sur le calcul de degré de dépression, étant donné sa popularité et son potentiel de prévalence dans la communauté, et surtout pour prévenir les tragédies qui pourraient survenir à la suite d'un tel trouble. Pour cela, nous avons voulu appliquer le nouveau paradigme de l'Intelligence Artificielle, à savoir l'apprentissage en profondeur pour concevoir, proposer et développer une nouvelle méthode d'estimer le degré de dépression.

Après avoir étudié l'apprentissage en profondeur et la plupart de ses techniques, ainsi qu'exploré les travaux liés aux calcul de degré de dépression, nous avons conclu que l'algorithme de l'auto-encoder n'était utilisé que dans le domaine des images, où il a connu un grand succès et l'a révolutionné. Dans notre étude, nous avons essayé d'appliquer cet algorithme au texte et nous avons cherché à l'adapter à notre problème afin de donner une nouvelle idée et solution qui aiderait les chercheurs à savoir si l'utilisation de cette nouvelle technologie conduira à un grand succès dans le domaine du traitement du langage naturel tout comme les images. Par conséquent, nous avons développé deux modèles qui sont principalement basés sur deux types de réseaux de neurones profonds, le réseau de neurones convolutifs « CNN » et une combinaison de CNN et d'auto-encodeur. Avant de commencer à entrainer ces modèles, nous avons préparé l'ensemble de données fourni par CLEF eRisk2019, cet ensemble de données est basé sur des publications en anglais de la plate-forme de réseau social Reddit. Tout d'abord, nous avons extrait les caractéristiques de notre ensemble de données en deux étapes, la première est le prétraitement et la seconde est la vectorisation avec l'utilisation de la méthode Word2vec, cela facilite notamment l'analyse sémantique des mots. Puis, nous avons commencé à appliquer notre modèle proposé qui a ensuite été entraîné. Pour faire nos prédictions, nous avons utilisé une méthode statistique spécifique, à l'aide de l'ensemble de donnée fourni par CLEF eRisk2020. Nous avons eu l'opportunité d'évaluer nos résultats à l'aide du système d'évaluation CLEF eRisk2020.

Nos résultats finaux ont été très favorables dont nous avons obtenu le deuxième meilleur score dans les métriques d'évaluation ACR et AHR pour le modèle CNN, le deuxième meilleur score dans la première métrique d'évaluation ACR pour le modèle Auto-encoder_CNN par

rapport les travaux de l'année dernière et des résultats excellents pour notre modèle CNN surpassant tous les modèles qui ont utilisé par l'équipe USDB [56] à la conférence CLEF eRisk2020.

Nos principales contributions de ce travail reposent sur les trois points suivants :

- ✓ Nous avons utilisé une méthode statistique distincte pour la prédiction.
- ✓ Nous avons combiné le modèle CNN avec le modèle Auto-encoder afin d'améliorer le processus d'extraction de caractéristiques et d'améliorer les performances du modèle.
- ✓ Nous avons analysé à travers différentes expériences les performances de deux modèles d'apprentissage en profondeur afin de fournir plus de perspectives et d'informations pour les recherches sur la dépression.

Malgré les résultats très prometteurs, nous avons rencontré de nombreuses difficultés dans ce travail que nous avons pu surmonter. Nous n'avions pas une bonne base dans le domaine de l'IA ce qui nous empêchait de procéder facilement et rapidement dans nos recherches. Nous avons également eu des difficultés à trouver des articles dans ce domaine car ils ne sont pas disponibles et limités.

De plus, nous avons constaté certaines limitations avec notre ensemble de données, où nous avons entraîné nos modèles sur un petit ensemble de données avec des grandes publications, ce qui est un inconvénient majeur de l'apprentissage en profondeur. En outre, cet ensemble de données n'est pas riche en contenu, il contient trop d'erreurs grammaticales et syntaxiques.

Afin de surmonter ces limitations et d'améliorer la qualité des modèles, nous proposerons en perspectives :

- ➢ Entraîner nos modèles sur des dataset plus volumineux et plus précises.
- ➢ Entraîner nos modèles à base de l'apprentissage en profondeur sur des serveurs plus puissants avec des plus d'époques.
- ➢ Utiliser nos modèles avec différentes maladies mentales, pas seulement la dépression. La seule partie manquante pour le prouver ce sont les ensembles de données dédiés spécifiquement à divers troubles mentaux.
- ➢ Utiliser nos modèles pour calculer le degré de dépression à partir différentes plates-formes de réseaux sociaux, non seulement Reddit.

- Utiliser nos modèles pour calculer le degré de dépression dans d’autres langues, car nous nous concentrons uniquement sur les publications en anglais.

Pour conclure, ce projet a été une expérience d'apprentissage très enrichissante, car il nous a fait prendre conscience que la dépression est une maladie grave qui doit être prise au sérieux. Nous avons découvert de nouvelles technologies inconnues ainsi que le domaine de l'analyse des sentiments. Cela nous a également permis d'acquérir de nouvelles compétences et de maîtriser celles existantes. Nous avons également appris à maîtriser le langage python ainsi que la plate-forme Google Colab et PyCharm que nous continuerons à utiliser dans nos futurs travaux.

Annexe

[Annexe n°1] : Beck's Depression Inventory

1. Sadness
0. I do not feel sad.
1. I feel sad much of the time.
2. I am sad all the time.
3. I am so sad or unhappy that I can't stand it.

2. Pessimism
0. I am not discouraged about my future.
1. I feel more discouraged about my future than I used to be.
2. I do not expect things to work out for me.
3. I feel my future is hopeless and will only get worse.

3. Past Failure
0. I do not feel like a failure.
1. I have failed more than I should have.
2. As I look back, I see a lot of failures.
3. I feel I am a total failure as a person.

4. Loss of Pleasure
0. I get as much pleasure as I ever did from the things I enjoy.
1. I don't enjoy things as much as I used to.
2. I get very little pleasure from the things I used to enjoy.
3. I can't get any pleasure from the things I used to enjoy.

5. Guilty Feelings
0. I don't feel particularly guilty.
1. I feel guilty over many things I have done or should have done.
2. I feel quite guilty most of the time.
3. I feel guilty all of the time.

6. Punishment Feelings
0. I don't feel I am being punished.
1. I feel I may be punished.
2. I expect to be punished.
3. I feel I am being punished.

7. Self-Dislike
0. I feel the same about myself as ever.
1. I have lost confidence in myself.
2. I am disappointed in myself.
3. I dislike myself.

8. Self-Criticalness
0. I don't criticize or blame myself more than usual.
1. I am more critical of myself than I used to be.
2. I criticize myself for all of my faults.
3. I blame myself for everything bad that happens.

9. Suicidal Thoughts or Wishes
0. I don't have any thoughts of killing myself.
1. I have thoughts of killing myself, but I would not carry them out.
2. I would like to kill myself.
3. I would kill myself if I had the chance.

10. Crying
0. I don't cry anymore than I used to.
1. I cry more than I used to.
2. I cry over every little thing.
3. I feel like crying, but I can't.

11. Agitation
0. I am no more restless or wound up than usual.
1. I feel more restless or wound up than usual.
2. I am so restless or agitated that it's hard to stay still.
3. I am so restless or agitated that I have to keep moving or doing something.

12. Loss of Interest
0. I have not lost interest in other people or activities.
1. I am less interested in other people or things than before.
2. I have lost most of my interest in other people or things.
3. It's hard to get interested in anything.

13. Indecisiveness
0. I make decisions about as well as ever.
1. I find it more difficult to make decisions than usual.
2. I have much greater difficulty in making decisions than I used to.
3. I have trouble making any decisions.

14. Worthlessness
0. I do not feel I am worthless.
1. I don't consider myself as worthwhile and useful as I used to.
2. I feel more worthless as compared to other people.
3. I feel utterly worthless.

15. Loss of Energy
0. I have as much energy as ever.
1. I have less energy than I used to have.
2. I don't have enough energy to do very much.
3. I don't have enough energy to do anything.

16. Changes in Sleeping Pattern
0. I have not experienced any change in my sleeping pattern.
1a. I sleep somewhat more than usual.
1b. I sleep somewhat less than usual.
2a. I sleep a lot more than usual.
2b. I sleep a Iot less than usual.
3a. I sleep most of the day.
3b. I wake up 1-2 hours early and can't get back to sleep.

17. Irritability
0. I am no more irritable than usual.
1. I am more irritable than usual.
2. I am much more irritable than usual.
3. I am irritable all the time.

18. Changes in Appetite
0. I have not experienced any change in my appetite.
1a. My appetite is somewhat less than usual.
1b. My appetite is somewhat greater than usual.
2a. My appetite is much less than before.
2b. My appetite is much greater than usual.
3a. I have no appetite at all.
3b. I crave food all the time.

19. Concentration Difficulty
0. I can concentrate as well as ever.
1. I can't concentrate as well as usual.
2. It's hard to keep my mind on anything for very long.
3. I find I can't concentrate on anything.

20. Tiredness or Fatigue
0. I am no more tired or fatigued than usual.
1. I get more tired or fatigued more easily than usual.
2. I am too tired or fatigued to do a lot of the things I used to do.
3. I am too tired or fatigued to do most of the things I used to do.

21. Loss of Interest in Sex
0. I have not noticed any recent change in my interest in sex.
1. I am less interested in sex than I used to be.
2. I am much less interested in sex now.
3. I have lost interest in sex completely

Bibliographie

[1] T. Nasukawa, and J. Yi, Sentiment analysis: Capturing favorability using natural language processing. Proceedings of the 2nd international conference on Knowledge capture. pp. 70-77, 2003.

[2] G. Anadiotis, Tendance 2020 : IA, la question cruciale des données et du hardware. URL :*https://www.zdnet.fr/actualites/tendance-2020-ia-la-question-cruciale-des-donnees-et-du-hardware-39898455.htm* , Jeudi 13 Février, 2020.

[3] E. Charniak. *Introduction to artificial intelligence*. Pearson Education India, 1985.

[4] T. Hastie, R. Tibshirani and J. Friedman, The elements of statistical learning: data mining, inference and prediction, vol. 9. *Proc. Annu.* Springer, 2017.

[5] E. Bisong. *Building Machine Learning and Deep Learning Models on Google Cloud Platform.* Berkeley: *Apress, 2019.*

[6] G. Ciaburro. *MATLAB for Machine Learning.* Packt Publishing Ltd, 2017.

[7] B. Belainine, "Classification supervisée de textes courts et bruités : application au domaine des médias sociaux. " Mémoire de master. Université du Québec À Motréal, Avril, 2017.

[8] C. V. Hee, "L'analyse des sentiments appliquée sur des tweets politiques: une étude de corpus." Mémoire de master. Université Bruxelles Belgique, 2013.

[9] B. Pang, L. Lee, and S. Vaithyanathan, *Thumbs up? Sentiment classification using machine learning techniques.* arXiv preprint cs/0205070, 2002.

[10] L. ALI, " SÉLECTION DES MOTS CLÉS BASÉE SUR LA CLASSIFICATION ET L'EXTRACTION DES RÈGLES D'ASSOCIATION." Mémoire de master. Université du QUÉBEC À TROIS-RIVIÈRES, JUIN, 2017.

[11] M. Abdelkader, "L'ANALYSE DU SENTIMENT UTILISANT LE DEEP LEARNING." Mémoire de master. Université de Dr. TAHAR MOULAY SAIDA, 2019.

[12] T. Taulli, *Deep Learning*, in *Artificial Intelligence Basics: A Non-Technical Introduction*. Apress: Berkeley, CA. p. 69-90, 2019.

[13] U. Kamath, J. Liu, and J. Whitaker. *Deep learning for NLP and speech recognition*. Springer International Publishing: Cham. Vol. 84, 2019.

[14] J. Bootcamp, La vraie différence entre Machine Learning & Deep Learning. URL: *https://www.jedha.co/blog/la-vraie-difference-entre-machine-learning-deep-learning*, 2020.

[15] H. Saleh, *The deep learning with PyTorch workshop: build deep neural networks and artificial intelligence applications with PyTorch*. Packt Publishing Ltd., 2020.

[16] N. Sharma, R. Sharma, and N. Jindal, "Machine Learning and Deep Learning Applications-A Vision," *Glob. Transitions Proc.*, vol. 2, no. 1, pp. 24–28, doi: 10.1016/j.gltp.2021.01.004, 2021.

[17] V. Mittal, Top 15 Deep Learning applications that will rule the world in 2018 and beyond. *URL: https://medium.com/@vratulmittal/top-15-deeplearning-applications-that-will-rule-the-world-in-2018-andbeyond-7c6130c43b01* , 2017.

[18] I. Goodfellow,Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[19] B. Abdelkrim and O. Amina, "Une approche Deep Learning pour l'analyse des Sentiments Sur Twitter. " Mémoire de master. Université de Djilali BOUNAAMA - Khemis Miliana, 2018.

[20] C. Hardy, " Contribution au développement de l'apprentissage profond dans les systèmes distribués. " Thèse de doctorat. Université de Rennes 1, 2019.

[21] M. Minsky and S. Papert. *An introduction to computational geometry*. *Cambridge tiass. HIT*, 1969.

[22] Y. Mercadier, "Classification automatique de textes par réseaux de neurones profonds : application au domaine de la santé." Thèse de doctorat. Université de Montpellier, 2020.

[23] A. Khan, A. Sohail, U. Zahoora, and A.S. Qureshi, *A survey of the recent architectures of deep convolutional neural networks*, vol. 53, no. 8. Springer Netherlands, 2020.

[24] A. Habba and O. Ishak, "La classification des images satellitaires par l ' apprentissage profonde (deep learning)." Mémoire de master. Université de Ahmed Draia - Adrar, 2019.

[25] C. Etienne, " Apprentissage profond appliqué à la reconnaissance des émotions dans la voix." Thèse de doctorat. Université de Paris Saclay, 2019.

[26] C. Rodrıguez-Pardo, "Personalised aesthetics assessment in photography using deep learning." Master of Science in Artificial Intelligence School of Informatics University of Edinburgh, 2018.

[27] M. Yani, *and al, Application of transfer learning using convolutional neural network method for early detection of terry's nail*. *Journal of Physics: Conference Series*. Vol. 1201. No. 1. IOP Publishing, 2019.

[28] G. Gelly, "Reseaux de neurones récurrents pour le traitement automatique de la parole." Thèse de doctorat. Université de Paris-Saclay, 2017.

[29] B. Hammer, "On the approximation capability of recurrent neural networks."

Neurocomputing, vol. 31, no. 1-4, pp. 107–123, 2000.

[30] G. Parascandolo. *Recurrent neural networks for polyphonic sound event detection*. Thèse de maîtrise, 2015.

[31] L. Phong, and W. Zuidema. "Compositional distributional semantics with long short term memory." *arXiv preprint arXiv: 1503.02510*, 2015.

[32] R. Kessler, "Traitement automatique d'informations appliqué aux ressources humaines." Avignon, 2009.

[33] F. Yvon, "Une petite introduction au Traitement Automatique des Langues Naturelles." *Conference on Knowledge discovery and data mining*. 2010.

[34] M. Latour, "Analyse de sentiments dans les textes économiques : un exemple d ' application chez ReportLinker," 2021.

[35] E. Cambria. "Affective computing and sentiment analysis. " IEEE Intelligent Systems. 31 (2), pp. 102–107, 2016.

[36] A. Ziani, "La recommandation via l'analyse d'opinions." Mémoire de master. Université de Badji Mokhtar - Annaba,2018.

[37] B. Liu. *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University press, 2015.

[38] C. Kihel, "Analyse des sentiments en utilisant l'apprentissage Profond: Cas de la langue Arabe." Mémoire de master. Université de Mohamed khider - Biskra, 2020.

[39] S. Behdenna, F. Barigou, and G. Belalem, "Document level sentiment analysis: A survey." *EAI Endorsed Transactions on Context-aware Systems and Applications.* vol. 4, no. 13, 2018.

[40] P. Sasikala and LMI. Sheela, "Comparative Study of Sentiment Analysis Techniques in Web." International Journal of Scientific & Engineering Research, vol. 8, no. 5, pp. 125–129, 2017.

[41] P. Kumar and UC. Jaiswal, "A Comparative Study on Sentiment Analysis and Opinion Mining." *Int J Eng Technol*, vol. 8, no. 2, pp. 938-943, 2016.

[42] M.D. Devika, C. Sunitha, A. Ganesh. Sentiment Analysis :A Comparative Study On Different Approaches. Fourth International Conference on Recent Trends in Computer Science Engineering, 2016.

[43] C. Hermann. Entre Web 2.0 et 3.0: opinion mining. Thèse de doctorat. Haute Ecole de Gestion & Tourisme, 2010.

[44] S. Maurel, P. Curtoni and L. Dini, L'analyse des sentiments dans les forums. *Atelier Fouille des Données d'Opinions (FODOP 08)*. pp. 9–22, 2008.

[45] D. Boullier and A. Lohard. *Opinion mining et Sentiment analysis: Méthodes et outils*. OpenEdition Press, 2012.

[46] F. Belbachir. " Expérimentation de fonctions pour la détection d'opinions dans les blogs. " Mémoire de master. Université de Paul Sabatier, Toulouse, 2010.

[47] R. BENKHELIFA and S. GAGUI, "Fouille de données d'opinion des usagers de sites E-commerce." Mémoire de master. Université de Kasdi Merbah - Ouargla, 2013.

[48] K. Bayingana and J. Tafforeau, " La Dépression Etat des connaissances et données disponibles pour le développement d'une politique de santé en Belgique," p. 116, 2002, [Online]. Available: https://www.wiv-isp.be/epidemio/epifr/crospfr/depression_fr.pdf.

[49] C. Ibrahim." Identifying Depression in Tweets using Deep Learning." Mémoire de master. Université de Saad Dahleb - Blida , 2018.

[50] F. Nacera, M. Boumediene, " DETECTION OF DEPRESSION USING E-RISK DATASET." Mémoire de master. Université de Saad Dahleb - Blida , 2019.

[51] D.E. Losada, F. Crestani, and J. Parapar. *Early detection of risks on the Internet: an exploratory campaign*. in *European Conference on Information Retrieval*. Springer, 2019.

[52] A.T. Beck, R.A. Steer, and M.G. Carbin. *Psychometric properties of the Beck Depression Inventory: Twenty-five years of evaluation.* Clinical psychology review, vol. 8, no 1, pp. 77-100, 1988.

[53] D. E. Losada, F. Crestani, and J. Parapar, "Overview of eRisk 2020: Early Risk Prediction on the Internet," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12260 LNCS, pp. 272–287, doi: 10.1007/978-3-030-58219-7_20, 2020.

[54] A.T. Beck and al. An Inventory for Measuring Depression. *Archives of general psychiatry*, vol. 4, no 6, p. 561-571, 1961.

[55] A. Trifan, P. Salgado, and J. L. Oliveira, "BioInfo@UAVR at eRisk 2020: on the use of psycholinguistics features and machine learning for the classification and quantification of mental diseases," pp. 22–25, 2020, [Online]. Available: http://early.irlab.org/.

[56] A. Madani, F. Boumahdi, and A. Boukenaoui, "USDB at eRisk 2020 : Deep learning models to measure the Severity of the Signs of Depression using Reddit Posts," pp. 22–25, 2020.

[57] R. Martínez-Castaño, A. Htait, L. Azzopardi, and Y. Moshfeghi, "Early risk detection of self-harm and depression severity using BERT-based transformers : iLab at CLEF eRisk 2020," pp. 22–25, 2020.

[58] AS. Uban and P. Rosso, "Deep learning architectures and strategies for early detection

of self-harm and depression level prediction." *CEUR Workshop Proceedings*. Sun SITE Central Europe, 2020.

[59] D. Maupom and al, "Early Mental Health Risk Assessment through Writing Styles , Topics and Neural Models." *CLEF (Working Notes)*.pp. 22–25, 2020.

[60] PaddlePaddle, licensed under the Creative Commons Attribution-Share in the Same Way 4.0 International License Agreement. 2020,[Online]. Available: https://github.com/PaddlePaddle/book/tree/develop/04.word2vec

[61] S. Meena, *Training Word2vec using genism*. Medium , 2020.

[62] I. Joosten, "Using word embeddings for outlet recommendation." Thèse de doctorat. Tilburg University, 2019.

[63] R. Hashempour, "From Word Vectors to Sentence Vectors," p. 68, 2018, [Online]. Available: https://lct-master.org/contents_2014/theses.php%0Ahttps://lct-master.org/getfile.php?id=3759&n=1&dt=TH&ft=pdf&type=TH.

[64] S. Amine and al., CLASSIFICATION SUPERVISÉE DE DONNÉES PÉDAGOGIQUES POUR LA RÉUSSITE DANS L' ENSEIGNEMENT SUPÉRIEUR. Thèse de doctorat. I3S, Université Côte d'Azur, 2020.