

UNIVERSITÉ DES SCIENCES ET DE LA
TECHNOLOGIE HOUARI BOUMEDIENE



La prédiction de personnalité sur les
réseaux sociaux

Binôme

BENDALI OMAR MANIL

DJEBRI MOUNIRA

20 juin 2020

Table des matières

1	Introduction Générale	7
2	Psychologie de la personnalité	11
2.1	Introduction	11
2.2	Psychologie de la personnalité	11
2.3	Théories de la personnalité	12
2.4	Modèles descriptifs de la personnalité	12
2.4.1	Modèle descriptif de la personnalité Big Five	13
2.4.2	Modèle descriptif de la personnalité MBTI	16
2.5	Conclusion	18
3	Corpus et apprentissage automatique	21
3.1	Introduction	21
3.2	Corpus d'apprentissage	21
3.2.1	myPersonality :	22
3.2.2	SinaMicroblog :	22
3.2.3	Essays :	23
3.2.4	MBTI Dataset :	23
3.2.5	TWISTY :	23
3.3	Travaux connexes	24
3.4	Traitement Automatique du Langage Naturel	24
3.4.1	Prétraitement	25
3.4.2	Nettoyage du bruit	25
3.4.3	Normalisation	26
3.4.4	Correction des fautes d'orthographe	26
3.4.5	Suppression de mots vides	26
3.4.6	Tokenization	26
3.4.7	Racinisation (Stemming)	27
3.4.8	Lemmatisation	27
3.5	Représentation du texte :	27
3.5.1	Pondération	27
3.5.2	Représentation 1 parmi n (One-Hot)	29
3.6	Plongement Lexical (Word Embedding)	29
3.6.1	Word2Vec	30

3.6.2	Vecteurs Globaux (GloVe)	35
3.6.3	FastText	37
3.7	Apprentissage Automatique	38
3.7.1	Apprentissage Supervisé	38
3.7.2	Apprentissage Non Supervisé	38
3.7.3	Apprentissage Semi-Supervisé	39
3.7.4	Méthodes d'apprentissage	39
3.8	Réseaux de Neurones Artificiels	44
3.8.1	Réseaux de Neurones Récurrents (RNN)	50
3.8.2	Réseaux Récurrents à Mémoire Court et Long terme (LSTM)	51
3.8.3	Réseaux de Neurones Récurrents à Portes (GRU)	53
3.8.4	Réseaux Récurrents Bi-directionnels (Bi-RNN)	54
3.8.5	Réseaux de Neurones Convolutifs (CNN)	55
3.9	Attaques Adversaires	56
3.9.1	Types d'attaques adversaires	57
3.9.2	Risques adversaires	58
3.10	Apprentissage Adversaire	58
3.10.1	Entraînement adversaire pour la classification de texte	60
3.10.2	Entraînement adversaire virtuel pour la classification de texte	61
3.11	Mesures de performance	62
3.11.1	Précision	62
3.11.2	Rappel	63
3.11.3	F-score	63
3.11.4	Exactitude (Accuracy)	63
3.11.5	Coefficient de Corrélation de Matthews (MCC)	63
4	Conception	65
4.1	Introduction :	65
4.2	Conception Générale du Système de Prédiction de Personnalité	65
4.3	Préparation du Corpus	66
4.3.1	Description du Corpus d'Apprentissage :	66
4.3.2	Prétraitement et Représentation du texte :	69
4.3.3	Plongement Lexical	74
4.4	Classifieur pour la prédiction des traits de personnalité	75
4.4.1	Réseaux récurrents	75
4.4.2	Choix des hyperparamètres	77
4.4.3	Hyperparamètres des réseaux récurrents :	77
4.4.4	Réseaux Convolutifs	79
4.4.5	Hyperparamètres du Réseau Convolutif :	80

5 Tests et Expérimentations	81
5.1 Introduction	81
5.1.1 Méthodologie	81
5.1.2 Environnement d'expérimentation	81
5.2 Tests comparatifs	82
5.2.1 Discussion des résultats	82
5.2.2 Comparaisons	83
5.2.3 Interpretations	84
6 Conclusion générale	87
Annexes	89
A Application	91
A.1 Scénario d'utilisation :	92
A.1.1 Passer le test de personnalité :	92
A.2 Collecte des données :	98
A.3 Prédiction de Traits de personnalité :	98
B Covid-19	101
Bibliographie	108

Introduction Générale

Qu'est-ce que la personnalité ? Comment comprendre les variations qui font la singularité de chacun ? Serions-nous tous bien plus similaires que nous le pensons ? Comment la personnalité influence-t-elle nos interactions avec les autres, voir même nos prises de décisions ? Existe t-il un moyen de caractériser la personnalité d'un individu ?

Ces questions ne sont pas nouvelles et restent encore aujourd'hui, pour certaines, sans réponse ou bien avec des réponses très partielles et non encore satisfaisantes. En effet, de tout temps, l'homme, en quête perpétuelle de réponses, tente d'interpréter les comportements et préférences de l'autre. On en retrouve les premiers jalons dans l'Antiquité.

Pour tenter de répondre à ces questions, il faut d'abord pouvoir décrire la personnalité. La psychologie humaine, jusqu'à récemment, perçue comme très complexe, peut être mieux appréhendée en étudiant l'ensemble des comportements qui constituent l'individualité d'une personne, autrement connue comme la personnalité.

En effet, la personnalité rend compte des différences comportementales d'une personne, ce qui la qualifie et fait sa singularité.

Le champ de la psychologie de la personnalité est particulièrement vaste, c'est une thématique qui a fait, et fait encore l'objet de nombreux travaux. Il existe actuellement presque autant de théories de la personnalité que d'auteurs qui ont abordé le sujet et qui n'ont pas toutes été validées ou comparées. Une des difficultés premières de ce projet est de choisir le(s) modèle(s) à utiliser pour notre étude. Pour préciser notre propos, il convient d'expliquer ce qu'on entend par théorie en psychologie humaine.

Une théorie de la personnalité est un ensemble de principes et de règles permettant de prédire, à propos d'un individu, certaines caractéristiques. La théorie permet en effet de faire des pronostics sur la façon dont un individu va se comporter dans différentes situations.

Parmi la pléthore de théories existantes, on distingue les théories basées traits, et celles basées types. La première proposant une classification de personnalité en caractéristiques constantes, la seconde quant à elle se veut plus englobante et propose une classification par ensemble de traits. Pouvoir décrire les éléments caractéristiques de la personnalité suppose la possibilité d'être en mesure d'évaluer ces caractéristiques. Les représentations de la personnalité humaine, sous forme de typologies en types ou en traits, ont permis la création d'outils de mesure pour décrire la personnalité, ou plutôt pour établir les variations quantitatives et qualitatives d'une caractéristique chez un individu. Ces outils sont divers et associés à des méthodologies propres. Un premier défi sera le choix de l'outil à utiliser, nous en présenterons deux : le Big Five[1] et le MBTI[2].

Par ailleurs, la singularité d'un individu ou sa personnalité peut être dépeinte de par sa façon de s'exprimer. De nombreux psychologues se sont tournés vers la langue comme source d'attributs afin de développer une taxonomie¹ scientifique des traits de la personnalité [3]. Le fait que les individus diffèrent dans la façon dont ils parlent et écrivent n'est pas une observation nouvelle. Même lorsque le contenu du message est le même, les individus s'expriment avec leurs propres styles distincts.

Soutenant l'hypothèse selon laquelle les traits de personnalité des individus sont implicitement encodés dans les mots utilisés pour construire une phrase, de nombreuses études ont lié l'utilisation de la langue à un large éventail de dimensions psychologiques confirmant ainsi que la façon dont nous nous exprimons reflète notre personnalité. Par exemple, les introvertis ont tendance à utiliser un lexique plus diversifié, des constructions plus élaborées mais des mots d'émotion moins positifs.

Cette approche lexicale du profilage associée au caractère quantifiable des tests de la personnalité, ont permis de systématiser la prédiction de traits de personnalité, offrant un nouveau champs d'études au domaine du traitement automatique du langage² (TAL) et à l'apprentissage automatique.

Dans ce contexte, les réseaux sociaux offrent une possibilité d'étude de la personnalité. Ils ont suscité l'intérêt des scientifiques des données ainsi que des psychologues qui voient en la reconnaissance des dimensions psychologiques sur les réseaux sociaux un moyen d'analyse à grande échelle.

D'autre part, Mitja D.Back confirme dans son étude [4] que les comportements sur les médias reflètent les agissements réelles des individus et non une version idéalisée comme certains pourraient se laisser à croire.

1. La taxonomie est la science de la classification.

2. Le traitement automatique de données extraites des médias sociaux doit arriver à déterminer les méthodes les plus appropriées pour l'extraction d'information.

Les réseaux sociaux ont pris une place prépondérante dans la vie quotidienne de chacun d’entre nous. Chaque seconde, des utilisateurs du monde entier partagent du contenu, exposant ainsi leurs activités, leurs opinions, leurs sentiments et leurs pensées, entraînant la création de vastes volumes de données. Ces informations concernant les individus sont si complètes qu’elles sont devenues essentielles pour les applications de profilage.

Il est à noter que le caractère abondant des données textuelles sur les réseaux sociaux ne garantit en rien leur accessibilité, le cadre juridique renforçant les droits des utilisateurs de ces plateformes rend l’extraction de ses données un défi de taille.

Connaître la personnalité d’un individu est un indicateur sur ses réactions notamment lorsqu’il fait face à différentes situations. Prédire la personnalité d’un individu est alors intéressant pour de nombreux domaines tels que la politique, l’économie ou encore la médecine. Par exemple, des chercheurs en santé publique ont identifié une multitude de corrélations entre la personnalité et la consommation d’alcool et de tabac (McAdams Donnellan[5], Mezquita et al. [6], Paunonen Ashton [7]), la fréquence de l’exercice physique (Rhodes Smith [8]), le cholestérol et les triglycérides (Sutin et al. [9]), la longévité (Friedman et al. [10], Roberts et al. [11]) et la santé mentale (Goodwin Friedman [12], Ozer Benet-Martinez [13]).

Motivés par le succès des réseaux de neurones dans un large éventail d’applications liés à la classification de textes, nous nous proposons à travers ce projet d’explorer les techniques d’apprentissage profond supervisé et semi-supervisé dans le cadre de la prédiction de traits de personnalité sur les réseaux sociaux. De plus, afin d’apporter une contribution nouvelle à ce domaine de recherche, nous aborderons une technique d’apprentissage dite adversaire qui est une méthode de régularisation pour les classificateurs afin d’améliorer leur robustesse au bruit. Technique largement appliquée dans le domaine de la catégorisation d’image, elle est introduite dans le domaine de la classification de texte par Goodfellow [14] qui arrive à des performances surpassant l’état de l’art. Ce projet se présente en 4 chapitres :

- Un premier chapitre qui présente les concepts de base de la psychologie de la personnalité et les modèles descriptifs de la personnalité les plus connus, car la compréhension de ces aspects est essentielle lors de la conception de notre solution.
- Le second chapitre sera consacré aux techniques de prétraitement des données textuelles ainsi qu’à l’apprentissage automatique et les méthodes qu’il offre pour la résolution de problèmes.
- Le troisième chapitre concerne la conception de notre solution.
- Le quatrième et dernier chapitre présentera les expérimentations, tests effectués et résultats obtenus.

A la fin du mémoire, dans l'annexe A, nous présentons les services que nous proposons au travers d'une application web. Ces services sont : un test de personnalité, une collecte de données et enfin une prédiction de traits de personnalité.

Nous vivons actuellement une période sans précédent, en quelques semaines nous avons dû nous adapter à un monde nouveau, régi par des règles qui viennent bouleverser nos interactions sociales, nos habitudes et nos comportements de façon général. Dans quel mesure notre singularité en tant qu'individu, nos traits de personnalité, rendent-ils compte de notre propension à adhérer ou pas à de nouvelles mesures ? Nous nous sommes penché sur la question, à la recherche d'une quelconque relation entre les traits de personnalité et la progression de la pandémie de COVID-19 dans le monde. Nous explorons cela en détails dans l'annexe B.

Psychologie de la personnalité

2.1 Introduction

Dans ce chapitre nous introduisons certains concepts de base de la psychologie de la personnalité ainsi que les différents modèles descriptifs de la personnalité qui sont essentiels pour la compréhension de notre travail, ainsi que pour une meilleure interprétation de nos résultats.

2.2 Psychologie de la personnalité

Les racines de la psychologie de la personnalité remontent à l'époque des Grecs Anciens. Theophrastus, disciple d'Aristote compose en 300 ans avant **Jésus - Christ** des croquis de personnages, chacun d'eux étant personnifié avec un trait de personnalité particulier. La première taxonomie de traits de personnalité est proposée par le physicien grec Gallen, établissant une distinction entre les tempéraments sanguins, colériques, mélancoliques, etc.

Ce n'est que pendant la première guerre mondiale que les psychologues inventent des tests de mesure des traits de personnalité afin de faciliter le processus de sélection des soldats[15]. Dans le même temps, les théoriciens en psychanalytique proposent une nouvelle vision de l'individualité humaine, en mettant l'accent sur l'aspect de l'inconscient humain dans l'interprétation du comportement humain.

Deux courants de pensée apparaissent donc :

- La psychométrie : qui accentue la quantification précise des traits communs mesurables par auto évaluation consciente, et se manifestant dans des comportements observables.
- La psychanalytique : qui met en évidence le caractère complexe et unique des individus.

2.3 Théories de la personnalité

Durant de nombreuses années, le domaine de la psychologie de la personnalité a été dominé par les théories du courant psychanalytique comme la théorie séminale de Freud [16] (1955) qui décrit tout comportement humain comme étant déterminé par des forces inconscientes sur lesquelles l'individu a peu de contrôle. D'autres théories apparaissent ensuite comme les théories comportementalistes (B.F Skinner)[17], qui ont une vision adaptative de la personnalité et la voient comme une adaptation à l'environnement.

Plus tard, un nouveau courant émerge : le courant psychométrique. Dans cette approche les théoriciens cherchent à décomposer la personnalité en dimensions afin de réaliser des typologies d'individus ou de pratiquer des regroupements. Ces théories distinguent un trait de personnalité d'un type de personnalité.

Un trait de personnalité est une caractéristique stable, qui dure dans le temps. Elle représente la tendance à se comporter d'une certaine façon face à différentes situations. Elle est présente chez tous les individus, toutefois son intensité (score) diffère d'une personne à une autre. Des traits habituels sont par exemple l'extraversion, la curiosité, ou encore l'empathie.

Un type, quant à lui, est une caractéristique générale qui englobe des caractéristiques plus spécifiques, telles que les traits. Un type est par définition discret, c'est à dire qu'une personne possède ou non cette caractéristique.

Selon les théories des types, les personnes introverties et extraverties font partie de catégories opposées. Contrairement aux théories des traits où ces caractéristiques font parties d'un ensemble continu (d'intensités différentes), des individus pouvant avoir un trait à intensité moyenne.

2.4 Modèles descriptifs de la personnalité

Afin de permettre la description de la personnalité, les chercheurs ont mis en avant plusieurs taxonomies, les plus utilisées sont le Modèle des Big Five[1] et le Myers Briggs Type Indicator (MBTI)[2].

2.4.1 Modèle descriptif de la personnalité Big Five

Après des années de recherche, un consensus semble se profiler sur une taxonomie générale des traits de la personnalité, celle des cinq grands facteurs de la personnalité, connus sous le nom des « Big Five » ou Modèle OCEAN.

Ce modèle considère que tous les traits de personnalité antérieurement décrits se rattachent à l'un des grands domaines suivants¹ : L'Ouverture à l'expérience (O), la Conscience (C), l'Extraversion (E), l'Agréabilité (A) et le Névrotisme (N).

Le modèle des cinq facteurs est une classification hiérarchique des traits de personnalité (McCrae et Costa [18]). Il a été construit selon une démarche inductive² [19], approche utilisée initialement par Allport et Odbert [20] (1937) et qui permet d'identifier les dimensions de la personnalité par le biais d'une stratégie taxonomique.

McDougall [21] (1932) fut l'un des premiers à affirmer que la personnalité pouvait s'expliquer à partir de cinq facteurs distincts. Quelques années plus tard, Cattell [22] (1946) en arrive à un constat similaire après avoir développé une taxonomie à 16 traits primaires et 5 traits secondaires, puis Norman [23] (1963) ainsi que McCrae et Costa [18] (1987).

En parallèle aux analyses factorielles³ visant à cerner la structure du comportement humain, une approche dite lexicale fit son apparition (Goldberg [3]).

L'hypothèse lexicale postule que la plupart des caractéristiques importantes et socialement pertinentes de la personnalité d'un individu sont encodées dans le langage naturel sous forme d'adjectifs.

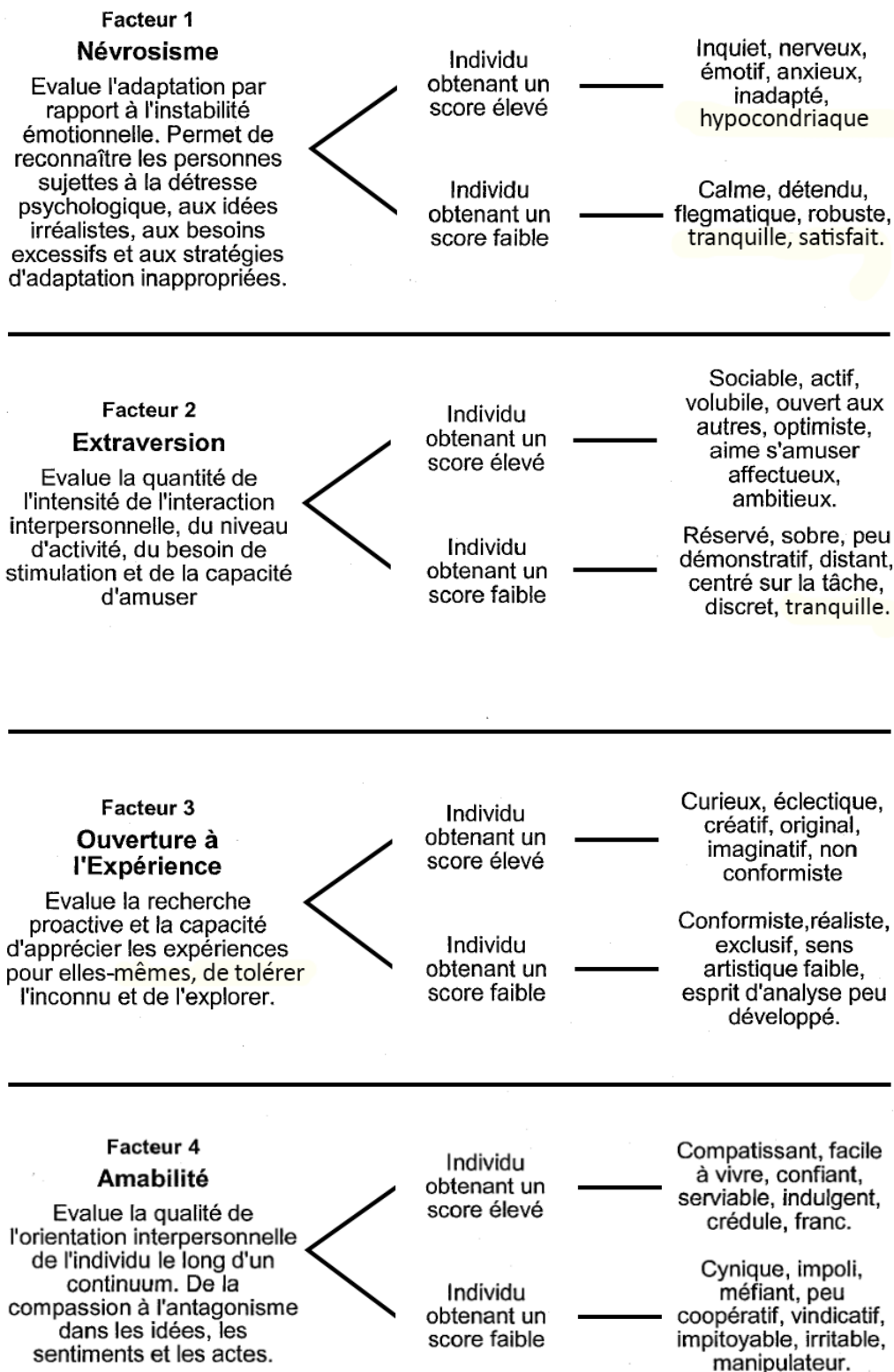
Les recherches réalisées dans le contexte de l'approche lexicale en sont également arrivées à un modèle de la personnalité basé sur cinq facteurs [3]. En somme, un modèle de la personnalité basé sur cinq grands facteurs émerge quelle que soit l'approche à laquelle recourent les chercheurs et des dimensions relativement semblables sont retrouvées.

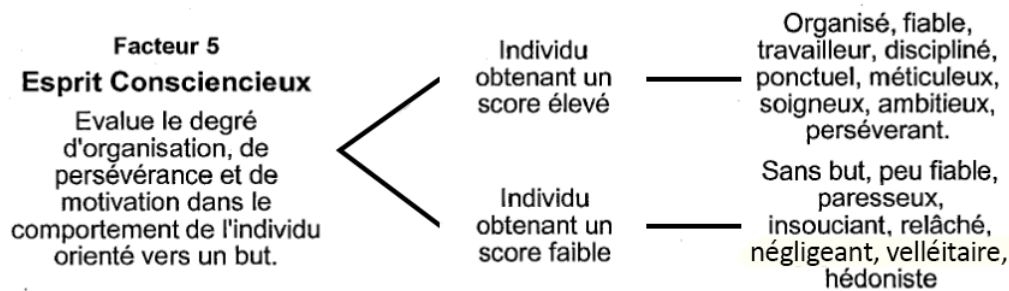
Voici la description des cinq traits :

1. Le nom des traits du modèle Big Five varie légèrement en raison du nombre important de sources qui ont contribué à sa conception. On retrouve généralement les variations suivantes : ouverture à l'expérience, ouverture d'esprit, conscience, conscience, extraversion, agréabilité, amabilité, convivialité et névrotisme, névrosisme.

2. manière de raisonner qui consiste à aller du singulier au général et des effets à la cause

3. méthode mathématique expérimentale, spécialement employée en psychologie, qui a pour objet l'étude des dimensions, ou facteurs, d'un domaine empirique donné.





(Source : Costa et McCrae, 1992; p. 2)

Le modèle des cinq facteurs, jouit d'un appui important de la part des psychologues de la personnalité, principalement car les cinq facteurs (ou traits) ont un caractère universel, ils se trouvent dans la majorité des cultures, et ne sont influencés ni par l'âge ni par le sexe. Par ailleurs, DeYoung et al. [24] trouvent des associations entre quatre des cinq grands traits et le volume des régions du cerveau qui y sont théoriquement associées. Par exemple, ils constatent que la conscience est associée au volume du cortex préfrontal latéral⁴, qui joue un rôle important dans la planification et le contrôle des impulsions.

2.4.1.a Outils de mesure de la personnalité

Les traits de personnalité sont généralement déterminés à l'aide d'un questionnaire, qui présente un nombre variable de questions qui décrivent des situations et des comportements courants, comme « Je tiens mes promesses » et l'individu doit indiquer dans quelle mesure chaque énoncé se décrit en utilisant une échelle de Likert⁵ en cinq points, allant de « fortement en désaccord » à « fortement d'accord ».

Le questionnaire de personnalité NEO, édition révisée (NEO-PI-R)[25] est sans nul doute l'instrument de mesure le plus fiable suivant la théorie des Big Five (Weiner Greene [26]). Il s'agit d'un test conçu par Costa et McCrae en 1992. Ce questionnaire peut être administré sous forme d'auto-évaluation (forme S) ou sous forme d'hétéro-évaluation par un observateur (forme R).

Il a été traduit dans plusieurs langues et de nombreuses recherches multiculturelles réalisées dans le cadre du modèle des Big Five y ont eu recours (McCrae[27]) démontrant ainsi sa validité transculturelle.

Ces tests, longtemps réservés aux applications cliniques et aux départements des ressources humaines sont aujourd'hui facilement accessibles en ligne, mis à disposition de tous par Goldberg [3] via International Personality Item Pool (IPIP)⁶

4. région du cerveau et siège de différentes fonctions cognitives comme le langage, la mémoire de travail, le raisonnement, etc.

5. outil psychométrique permettant de mesurer une attitude chez des individus.

6. Goldberg propose de mettre à la disposition du public un ensemble de questionnaires de la personnalité créant ainsi l'IPIP (www.ipip.ori.org)

2.4.2 Modèle descriptif de la personnalité MBTI

La classification des types psychologiques est une théorie de la personnalité développée par le psychiatre C. G. Jung[28] pour expliquer les différences psychologiques. Sur la base de ses observations, Jung conclut que les différences de comportement sont le fruit des tendances spontanées qui poussent chacun à fonctionner de manière différente. L'approche *Jungienne* met en évidence trois dimensions de base de la personnalité. Pour chacune d'elles sont définies deux façons opposées de fonctionner : l'orientation de l'énergie, soit introvertie (I), soit extravertie (E) ; la perception avec une dominante soit intuition (N), soit sensation (S) et le jugement privilégiant soit la pensée (T), soit les sentiments (F).

Une quatrième dimension a été ensuite formalisée par Myers[29] à partir de la théorie de Jung : le style de vie, les sujets s'inscrivant, en fonction de leurs préférences, soit dans une logique de maîtrise de leur environnement pour le jugement (J), soit dans une logique d'adaptation aux circonstances pour la perception (P).

Extraversion (E) / Introversion (I)	Sensation (S) / Intuition (N)
(E) : orientation vers le monde extérieur. (I) : orientation vers le monde intérieur.	(S) : percevoir les événements ou les détails du moment présent. (N) : percevoir les possibilités et les intuitions de l'avenir.
Pensée (T) / Sentiment (F)	Jugement (J) / Perception (P)
(T) : utiliser raison et la logique. (F) : se baser sur les valeurs personnelles et les émotions.	(J) : le contrôle, l'organisation et la structuration. (P) : la spontanéité et la flexibilité.

Source : (Myers et Kirby, 1994)

FIGURE 2.1 – Les quatre fonctions de Jung

La combinaison de ces 4 composantes permet de distinguer 16 types (portraits psychologiques) de personnalité, qui sont :

<p>ISTJ Administrateur Calme, méthodique, réservé, observateur, et responsable.</p> <p>ISFJ Protecteur Bon observateur, aimant l'harmonie, rendre service et éviter les conflits, tendance à devenir dépendant des autres.</p>	<p>ISTP Praticien Calme, observateur, réservé et imprévisible, réagissant en fonction de l'intérêt qu'il éprouve.</p> <p>ISFP Conciliateur Dévoué et gentil, observe et agit sans en parler, il aime la liberté dans un environnement de qualité.</p>
<p>ESTP Pragmatique Réaliste, réagissant rapidement aux situations. Bon observateur et a le sens de l'adaptation. Il aime l'action et la fête.</p> <p>ESFP Boute en Train Spontané, généreux et sympathique, prend les choses comme elles viennent pour en tirer le meilleur.</p>	<p>ESTJ Organisateur Franc, rapide, efficace et humain. Logique, se base sur le concret, aime les procédures et les plans d'action.</p> <p>ESFJ Nourricier Conformiste et discipliné. Cherchant à aider les gens de façon pratique.</p>
<p>INFJ Visionnaire Créatif, méthodique et organisé, sociable, mais solitaire à la fois, aimant l'harmonie.</p> <p>INFP Zélateur Idéaliste, créatif et très exigeant, fidèle à son non-conformisme. Il peut tomber dans le fanatisme.</p>	<p>INTJ Perfectionniste Innovateur et indépendant. Peut avoir tendance à être agressif et distant.</p> <p>INTP Concepteur Recherche sans cesse la cohérence des systèmes sans chercher à imposer ses opinions. Ses résultats ont peu de valeur à ses yeux.</p>
<p>ENFP Communicateur Créatif, curieux, anticipe les événements et éprouve du mal à décider.</p> <p>ENFJ Animateur Sens de la compréhension et de l'empathie, à l'aise à prendre la parole. Il peut avoir tendance à ne pas savoir dire non.</p>	<p>ENTP Innovateur Créatif et original, mais il s'ennuie dès qu'il s'agit de réalisation. Il a tendance à devenir versatile</p> <p>ENTJ Meneur Clair, direct et franc. Il a le sens du leadership et du défi, mais sa préférence pour le contrôle peut tourner à la domination.</p>

Source : (Myers et Kirby, 1994)

FIGURE 2.2 – Les caractéristiques des 16 types (MBTI)

La typologie de Jung appuyée par Myers donne naissance, en 1985, à un instrument d'auto-évaluation du nom de MBTI (Myers-Briggs Type Indicator).

Cet outil de mesure est composé de 100 items (propositions), axés principalement sur « comment le répondant se sent habituellement » et « comment il réagit dans des situations particulières » (Myers et Kirby [30]). Tel que : « Je vois les choses dans le détail, je suis toujours minutieux. » le sujet doit alors répondre sur une échelle en cinq points, allant de « fortement en désaccord » à « fortement d'accord ».

Bien que le MBTI soit un outil populaire et fréquemment utilisé, notamment par les compagnies lors des recrutements, sa validité reste encore à prouver. Sa fiabilité est très faible, étant donné que les réponses du participant dépendent grandement de son état au moment où il passe le test, en effet le MBTI est issue de la théorie des types qui ne sont pas, contrairement aux traits des attributs, stables. Des études ont montré que près de 58% des participants obtenaient des scores différents après avoir repassé le test au bout de 5 semaines[31].

2.5 Conclusion

Les avancées remarquables de la psychologie de la personnalité, ont contribué au renouvellement profond de multiples domaines de recherche tels que la santé publique, l'économie, l'éducation, la sociologie et la psychologie clinique. Soulignant largement la fiabilité des modèles descriptifs de la personnalité et plus particulièrement le modèle des Big Five (modèle des cinq facteurs).

Différentes études ont en effet prouvé la corrélation entre les traits de personnalité et certains types de comportement. Cela comprend entre autres les performances professionnelles. On pourrait poursuivre, en citant Ployhart et Bliese [32] qui suggèrent que les dimensions de la personnalité influencent significativement la manière dont les gens perçoivent et répondent aux changements et que certains traits seraient des antécédents particulièrement importants lorsque que l'on s'intéresse au potentiel et performance futurs d'un employé.

Des corrélations significatives ont également été trouvées entre la personnalité et les préférences d'un individu [33], permettant ainsi aux spécialistes du marketing d'y voir un prédicteur significatif de la réceptivité générale aux publicités, surpassant même le ciblage démographique.

Il est aussi intéressant de noter que le domaine de la politique s'est aussi approprié le sujet, théorisant sur les promesses de la prédiction de personnalité sur la sphère politique, y compris sa capacité à influencer le cours d'une campagne électorale. S'inscrivant dans cette tendance Jost et al. [34] ont démontré que la connaissance des traits de personnalité d'un individu permet de prédire s'il serait plus susceptible de voter pour McCain ou Obama à l'élection présidentielle de 2008 aux États-Unis.

Corpus et apprentissage automatique

3.1 Introduction

Les données textuelles sont complexes et requièrent un traitement minutieux, ceci est encore plus vrai pour les données textuelles des réseaux sociaux. Ces particularités dans les données soulèvent des défis quant aux méthodes de prétraitement à appliquer afin de les nettoyer, ainsi qu'à la manière de les représenter. Nous présenterons dans ce chapitre les différentes techniques de prétraitement des données en traitement automatique du langage naturel (TALN). Nous aborderons par la suite les méthodes d'apprentissage automatique pour la classification de texte.

3.2 Corpus d'apprentissage

Un corpus est un ensemble de documents regroupés, prêts à l'usage pour les tâches de traitement du langage naturel ou d'apprentissage automatique. Ainsi on peut les retrouver pré-étiquetés ou non. Étiqueter un texte, c'est lui associer une ou plusieurs catégories (étiquettes) selon la thématique qu'il aborde.

Le jeu ou données d'apprentissage (en anglais « dataset ») est un élément essentiel au processus d'apprentissage : si cet ensemble est inconsistant, le modèle d'apprentissage peut être de mauvaise qualité, un bon étiquetage est donc primordial au bon fonctionnement du système de classification automatique.

Les principaux corpus présents dans la littérature portant sur la prédiction de la personnalité sont :

3.2.1 myPersonality :

myPersonality est un projet de psychométrie mené entre 2007 à 2012 par l'Université de Cambridge, par Michael Kosinski et David Stilwell. Il consiste en une application Facebook qui administre un test de personnalité et collecte un large éventail d'informations personnelles et d'activités à partir du profil Facebook de l'utilisateur. En pratique, les participants ont été invités à répondre à un ensemble de questions (test dérivé du Costa et McCrae NEO-PI-R qui se base sur le modèle des cinq facteurs) puis avec leurs accords, l'application recueille l'ensemble de leurs publications Facebook. Ce corpus comportant les posts Facebook de près de 8 millions d'utilisateurs, est souvent considéré comme corpus de référence (ou Gold Standard). L'intégralité de ce corpus n'est toutefois plus disponible. Les chercheurs ont néanmoins mis à disposition un échantillon du corpus *myPersonality* original. L'échantillon comporte 9913 mises à jour de statut Facebook, en langue anglaise, de 250 utilisateurs (anonymisées), annotées avec le score du test de personnalité.

3.2.2 SinaMicroblog :

SinaWeibo, est un réseau social chinois de microblogging, permettant aux utilisateurs de poster de courts messages composés de photos ou de textes de 140 caractères maximum, puis de les commenter et de les partager avec leurs lecteurs. À l'image de Twitter, chaque utilisateur peut souscrire au contenu d'autres utilisateurs afin de recevoir leurs messages et mises à jour. Il est à noter que les spécificités de l'écriture chinoise rendent possible l'écriture d'un court texte en 140 caractères, alors qu'une telle longueur autorise seulement une courte phrase dans une écriture utilisant un alphabet latin.

Le jeu de données SinaMicroblog est alors collecté sur le service de microblog Weibo par Nie Dong en 2014, il s'agit d'un échantillonnage aléatoire de messages (random sampling) effectué quotidiennement sur un panel d'environ 1792 utilisateurs. Toutefois, seuls 994 utilisateurs ont consenti à passer le test de personnalité du Big Five, ce qui en fait un corpus partiellement étiqueté.

3.2.3 Essays :

Le Dataset de James Pennebaker et Laura King[35] a la particularité d'être l'un des premiers corpus reposant sur une approche psychométrique. Collecté entre 1997 et 2004, il contient 2 468 essais « libres » produits par des étudiants en psychologie. Les contributions écrites prenaient la forme d'un devoir de cours, mais n'étaient pas notées. Pour chaque devoir, les étudiants devaient écrire au moins 20 minutes par jour sur un sujet spécifique. Chaque élève rendait sa rédaction au bout 10 jours d'écriture. Par la suite, les essais ont été associés aux scores de personnalité des élèves, évalués en répondant au Big Five Inventory (BFI) [36].

3.2.4 MBTI Dataset :

En raison des coûts d'étiquetage élevés et les problèmes de confidentialité, le peu de corpus disponibles gratuitement sont peu diversifiés et de taille très réduite. Afin de palier à ce problème Matej Jurkovic and Jan Šnajder[37], collectent en 2018, les messages publiés dans le réseau social Reddit¹. Plus particulièrement, ils ne collectent que les messages des utilisateurs ayant précisé dans leur profil leurs indicateurs MBTI.

Ce dataset contient 354,996 messages (totalisant 921,269 mots) publiés par 13,631 utilisateurs. Ce qui en fait le plus grand ensemble de données étiquetées sur la personnalité disponible en termes de nombre de mots.[37]

3.2.5 TWISTY :

Twisty[38] est un ensemble de données populaire dans le domaine de la prédiction de personnalité. Ce corpus multilingues, est composé de 1,2 million de tweets de 18,168 utilisateurs s'étant identifié avec leurs indicateurs de type Myers-Briggs (MBTI).

A l'instar de *MBTI Dataset*, l'indicateur de type de personnalité (MBTI) est spécifié par l'utilisateur, les chercheurs partent donc du postulat que les utilisateurs ont passé un test de personnalité valide, et qu'ils reportent le résultat comme il a été présenté, sans aucune altérations.

1. Reddit est un gigantesque forum composé d'une multitude de sous-forums, où les internautes (330 millions d'utilisateurs actifs mensuels) partagent du contenu

3.3 Travaux connexes

La reconnaissance automatique des traits de personnalité à partir de données textuelles fait l'objet de nombreux travaux que ce soit sur les données de réseaux sociaux ou des données textuelles simples. Les premières tentatives de prédiction de personnalité se basaient essentiellement sur les Modèle à Vecteur de support (SVM) par l'utilisation de caractéristiques lexicales et syntaxiques. Néanmoins, les approches par apprentissage profond sont de plus en plus explorées ces dernières années. Kalghatgi et al. [39] proposent une approche utilisant un réseau de neurones multicouches avec pour entrée un corpus composé de caractéristiques syntaxiques (tel que le nombre moyen de mots positifs et négatifs) et de comportements sociaux (Nombre moyen de hashtags, etc.) extraits à partir de tweets. Dans une approche plus élaborée, Majumder et al. [40] utilisent un réseau de neurones convolutif afin d'extraire des vecteurs de caractéristiques textuelles qu'ils combinent à des informations extraites de texte brut à l'aide de la librairie Mairesse (qui contient une catégorisation de mots selon les traits) [41]. Face aux limitations quant à l'existence de corpus, de nombreux chercheurs ont été contraints de constituer leur propre jeu de données. Gjurovic et Snajder [37] proposent la construction d'un corpus en langue anglaise et d'un modèle de prédiction de personnalité construit à partir des données de plus de 9k utilisateurs incluant leur résultat au test de personnalité MBTI sur le réseau social Reddit. Similairement, TWISTY est un corpus multilingue (6 langues) pour le profilage du genre et de la personnalité proposé par Verhoeven [38] où les textes sont issus de Twitter et annotés avec les types de personnalités MBTI et le genre des auteurs. Également établi à partir de résultats de test MBTI, Personae est un corpus contenant du texte issu de 145 auteurs recueilli par Luyckx et Daelemans [42] qui proposent aussi un modèle pour la prédiction basé sur l'algorithme KNN. Face à la difficulté de construction d'un corpus annoté Nie et al. [43] ont une approche différente, et proposent un algorithme de régression linéaire local semi-supervisé pour des données extraites du réseau social Sina Weibo avec peu d'instances annotées et de nombreuses non annotées.

3.4 Traitement Automatique du Langage Naturel

Les réseaux sociaux intègrent un volume et une variété sans précédent de données textuelles. Toutefois l'étude des messages publiés par les internautes, qui sont par nature complexes, représente un nouveau défi pour le traitement automatique du langage (TAL).

En effet, les messages échangés sont rédigés de manière informelle, sur le ton de la conversation, et ressemblent plus à l’expression d’un « état d’âme » qu’à un travail réfléchi et révisé avec le soin habituellement attendu d’un texte formel. Les outils du TAL conçus pour les données traditionnelles se heurtent, par exemple, à l’emploi irrégulier, voire l’omission, de la ponctuation et des majuscules, tout comme la répétition (par exemple : *byebye*), l’élongation (par exemple : *coooooo*) [44], la contraction (par exemple : *I’ll* au lieu de *I will*), l’orthographe incorrecte et la multiplication d’abréviations populaires. Tous ces bruits compliquent les tâches essentielles au TAL telles que la segmentation.

Un autre obstacle à l’analyse syntaxique est la grammaire, ou pour être exacte son absence dans les médias sociaux [45](Kong et al., 2014). En effet, les phrases fragmentées sont devenues la norme à défaut des phrases complètes. Une adaptation des outils traditionnels s’avère donc nécessaire pour prendre en compte ces nouvelles variations.

3.4.1 Prétraitement

Le prétraitement des données est une étape essentielle pour l’amélioration de la qualité des données. Des données qui ne sont pas prétraitées ou brutes peuvent produire des résultats biaisés. Le prétraitement est un maillon important en traitement automatique du langage naturel. En effet, les données textuelles sont complexes et requièrent un traitement minutieux, ceci est encore plus déterminant pour les données textuelles des réseaux sociaux où le respect des conventions d’écriture y est rare, les erreurs orthographiques très fréquentes et les mots d’argot omniprésents.

Ces particularités dans les données soulèvent des défis quant aux méthodes de prétraitement à appliquer afin de nettoyer nos données, ainsi qu’à la manière de représenter les données textuelles.

3.4.2 Nettoyage du bruit

Contrairement aux corpus de textes traditionnels, les messages échangés dans les réseaux sociaux contiennent une grande quantité d’émoticônes et de symboles.

Les données “bruitées” et non structurées affectent considérablement les performances des outils TAL. Il s’agit donc d’éliminer tout symbole qui ne correspond pas à une lettre de l’alphabet (points, virgules, traits d’union, chiffres, etc.). Cette opération est motivée par le fait que ces caractères ne sont pas liés au contenu des messages et ne change rien au sens s’ils sont omis ; par conséquent, ils peuvent être négligés.

3.4.3 Normalisation

Une abréviation, forme abrégée d'un mot, des mots issus de l'argot ou discours familier peuvent induire un mauvais apprentissage du modèle, "influençant" et "biaisant" l'apprentissage. Plus spécifiquement, ces formes non standardisées peuvent engendrer des dissimilarités entre des messages comportant le même contenu.

L'usage répandu pour gérer ces mots consiste à les convertir en langage formel, les normalisant.

La normalisation fait donc référence au processus de transformation d'une forme langagière vers sa forme standard.

3.4.4 Correction des fautes d'orthographe

Une étape facultative du prétraitement consiste à corriger les mots mal orthographiés. En effet certaines représentations telles que le word embedding (concept introduit en 2.5), ont l'inconvénient majeur de ne pouvoir représenter les mots qui n'ont pas été observés durant la période d'apprentissage. Pour éviter donc d'avoir un pourcentage trop important de mots non reconnus nous essayons de les corriger à l'aide d'outil de correction automatique.²

3.4.5 Suppression de mots vides

Les mots vides sont tous les mots qui sont trop fréquents (ils n'aident donc pas à distinguer entre les documents) ou jouent un rôle purement fonctionnel dans la construction des phrases (articles, prépositions, etc).

Le résultat de la suppression des mots vides est que le nombre de mots dans la collection, ce qu'on appelle alors masse des mots, est réduit en moyenne de 50%[46]. Les mots à éliminer, connus comme stopwords, sont récoltés dans la stop liste qui contient en général entre 300 et 400 éléments en langue anglaise.

3.4.6 Tokenization

Pour mettre en œuvre des méthodes de catégorisation, il faut choisir un mode de représentation du texte, car les méthodes d'apprentissage ne sont toujours pas capables de représenter directement des données non structurées (texte brut).

2. Par exemple (Cucerzan et Brill, 2004) signalent que les fautes d'orthographe apparaissent dans jusqu'à 15% des requêtes de recherche sur le Web

La tokenisation, fait partie du processus de normalisation qui consiste en une segmentation ou ‘atomisation’ du texte en unités linguistiques manipulables (couramment nommés tokens) comme les mots, la ponctuation, les nombres, les symboles, etc.

3.4.7 Racinisation (Stemming)

L’étape du stemming ou racinisation consiste à remplacer chaque mot par sa racine comme par exemple les mots : rationnel, rationalité et rationnellement, sont remplacés par leur racine «rationnel» et les verbes conjugués par leur infinitif (rationalisèrent devient « rationaliser»). La racinisation n’a aucun impact sur la masse des mots (nombre des mots), mais réduit de 30% en moyenne la taille d’un message[46].

3.4.8 Lemmatisation

Le processus de lemmatisation consiste à utiliser des règles grammaticales pour remplacer les mots par leurs formes canoniques. Les lemmes correspondent donc à la forme des mots du dictionnaire ; par exemple, la forme canonique d’un verbe est l’infinitif, et celle d’un adjectif est au masculin singulier.

3.5 Représentation du texte :

La majorité des algorithmes d’apprentissage ne prennent pas en entrée du texte brut, mais des vecteurs numériques. Pour cela il est nécessaire de trouver une transformation représentative qui convertit le texte vers des vecteurs numériques.

La représentation la plus simple des documents textuels est appelée «représentation en sac de mots » (Bag of words, BoW) [47]. Elle consiste à transformer des textes en vecteurs où chaque élément représente une unité linguistique.

3.5.1 Pondération

Dans la littérature, les méthodes de transformation du texte vers des Bag-of-Words peuvent être divisées en trois approches principales. La plus utilisée est une approche purement statistique basée sur l’occurrence des termes comme TF(Term Frequency) et TF-IDF(Term Frequency-Inverse Document Frequency).

3.5.1.a TF-IDF

Les termes les plus fréquents ne sont pas nécessairement les plus pertinents. Au contraire, les termes qui apparaissent fréquemment dans un petit nombre de messages mais rarement dans d'autres tendent à être plus pertinents et spécifiques pour ce groupe particulier de message. Afin de capturer ces termes et de refléter leur importance, le schéma de pondération TF-IDF (fréquence des termes et inverse de la fréquence des documents) est très utilisé [48] :

$$W_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

tf (TermFrequency) : représente le nombre d'occurrences du terme dans le corpus.

$IDF = \log\left(\frac{N}{df_x}\right)$: permet de mesurer l'importance d'un mot.

N : le nombre de textes.

df_x : représente le nombre de textes contenant le terme.

Ces deux concepts sont combinés (par multiplication), en vue d'attribuer un plus fort poids aux termes qui apparaissent souvent dans un message et rarement dans l'ensemble du corpus.

Inconvénients : Les messages plus longs ont typiquement des poids plus forts parce qu'ils contiennent plus de mots, auquel cas « TF » tend à être plus élevé.

3.5.1.b TF-IDF-CF

Pour pallier aux lacunes de TF-IDF, un nouveau paramètre CF (class frequency) ou « fréquence de classe » a été introduit, celui-ci calcule la fréquence d'un terme dans une classe définie comme étant les groupes distinguables de messages.

$$TF.IDF.CF = tf_{x,y} \times \log\left(\frac{N}{df_x}\right) \times \frac{n_{cx,y}}{N_{cx}}$$

où c_{xy} représente le nombre de messages de la classe C à laquelle appartient y , où le terme x apparaît.

N_c Représente le nombre de messages de la classe C à laquelle appartient y .

3.5.2 Représentation 1 parmi n (One-Hot)

La méthode de représentation 1 parmi n (one-hot) est une méthode de représentation numérique de données textuelles. Elle consiste à représenter chaque mot du vocabulaire d'un corpus comme un vecteur unique binaire.

Exemple : Soit un vocabulaire V composé de : ['he', 'told', 'us', 'a', 'very', 'exciting', 'story'] La représentation au format one-hot d'un mot, est un vecteur de $|V|$ dimensions, dont tous les éléments sont nuls, hormis le composant du vecteur à la position du mot dans le vocabulaire. Pour le mot "story", sa représentation au format one-hot sera comme suit :

$$\text{story} \rightarrow \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

3.6 Plongement Lexical (Word Embedding)

Le plongement lexical, ou en anglais «Word Embedding», est une méthode d'apprentissage automatique permettant de générer une représentation numérique (vecteur à valeurs réelles) d'un vocabulaire. Cette méthode s'inspire de la théorie linguistique de Zellig [49], selon laquelle un mot serait défini par son contexte. Cette idée d'apprentissage de contexte permet de représenter deux mots étant utilisés dans le même contexte comme étant proches, contrairement à une représentation One-Hot où cet aspect est inexistant, comme le suggère la figure 3.1. Le plongement lexical permet également de capturer les relations sémantiques et syntaxiques entre les mots.

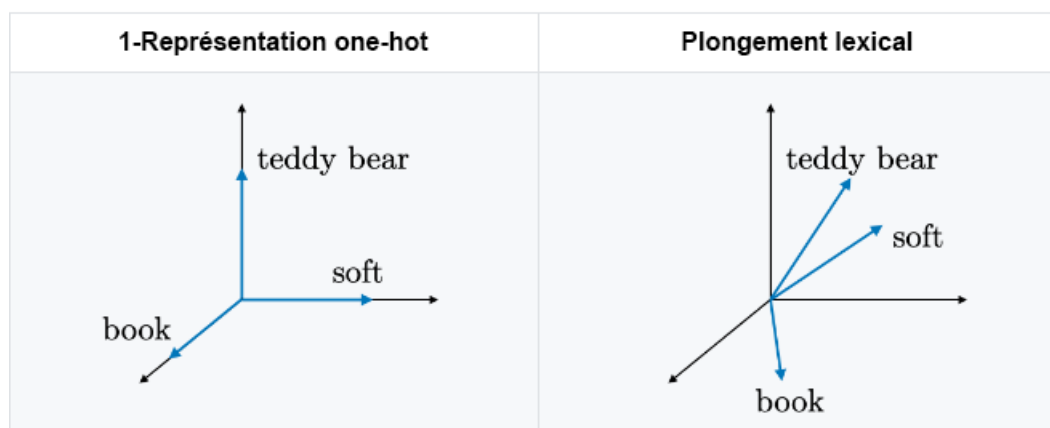


FIGURE 3.1 – Comparaison des représentations spatiales

Il existe plusieurs approches de plongement lexical, les plus performantes étant celles basées sur les réseaux de neurones.

3.6.1 Word2Vec

Word2Vec est un ensemble de méthodes de plongement lexical développées par Mikolov[50]. Les deux méthodes proposées sont : Sac de mots continu (CBOW : continuous bag of words) et Skip-Gram.

3.6.1.a Skip-Gram

Le modèle Skip-Gram est un réseau de neurones permettant de générer la représentation (plongement lexical) d'un mot avec en entrée le mot central et en sortie un des mots de son contexte représentant la cible.

Elle repose sur la modélisation de probabilité conditionnelle d'observer un mot w_i sachant qu'un mot w_j apparaît dans le contexte de w_i c'est à dire à l mots ou moins à droite ou à gauche[51].

$$p(w_j|w_i; U, V) = \frac{1}{1 + e^{-u_j^\top v_j}} \quad (3.1)$$

ou :

c : Le corpus de vocabulaire de taille N .

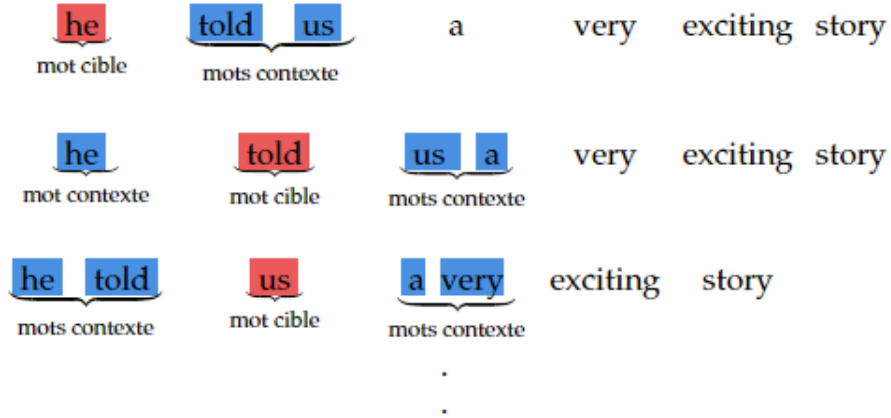
U : La matrice représentant les mots cibles, avec pour chaque mot w_i son vecteur ligne u_i le représentant.

V : La matrice représentant les mots contextes, avec pour chaque mot w_j son vecteur ligne v_j le représentant.

On génère ainsi pour une taille de fenêtre donnée l , un jeu de données D , un ensemble de couples (w_i, w_j) , où w_i représente le mot central et w_j un des mots du contexte de w_i .

Exemple :

Soit la phrase 'he told us a very exciting story', une taille de fenêtre $l = 2$, la génération du jeu de données sera effectuée en parcourant la phrase comme suit :



On obtient alors les couples du jeu de données suivants :

(he, told) (he, us)
 (told, he) (told, us) (told, a)
 (us, he) (us, told) (us, a) (us, very)
 .
 .
 .

Chaque couple du jeu de données sera une instance (un exemple) utilisée lors de l'apprentissage.

L'objectif de l'apprentissage sera de maximiser la vraisemblance de D :

$$\max_{U,V} \prod \sigma(u_j^T v_j) \quad (3.2)$$

avec $\sigma(x) = \frac{1}{1+e^{-x}}$

Maximiser la fonction 3.2 revient à maximiser son logarithme (le logarithme étant une fonction strictement croissante)

$$\max_{U,V} \log\left(\prod_{(w_i, w_j) \in D} \sigma(u_j^T v_j)\right) = \max_{U,V} \sum_{(w_i, w_j) \in D} \log \sigma(u_j^T v_j) \quad (3.3)$$

Calcul du plongement lexical :

L'entrée du réseau de neurones sera représentée par le mot cible sous son format one-hot, la sortie sera le mot contexte (également au format one-hot). Ce réseau possède une unique couche cachée de dimensions $|V| \times E$, avec E la dimension du plongement lexical. Les poids de ce réseau peuvent être représentés par deux matrices de poids $W1$ et $W2$, qui seront utilisées pour le calcul des plongements lexicaux. L'obtention du plongement lexical d'un mot se fait en multipliant la représentation du mot sous le format one hot à la matrice $W1$ (dans certaines implémentations à $W2$) obtenue après la fin de l'entraînement, comme le montre la figure 3.2. Cette opération permet de représenter le mot sous la d'un vecteur de E dimensions ($E=300$) à valeurs réelles.

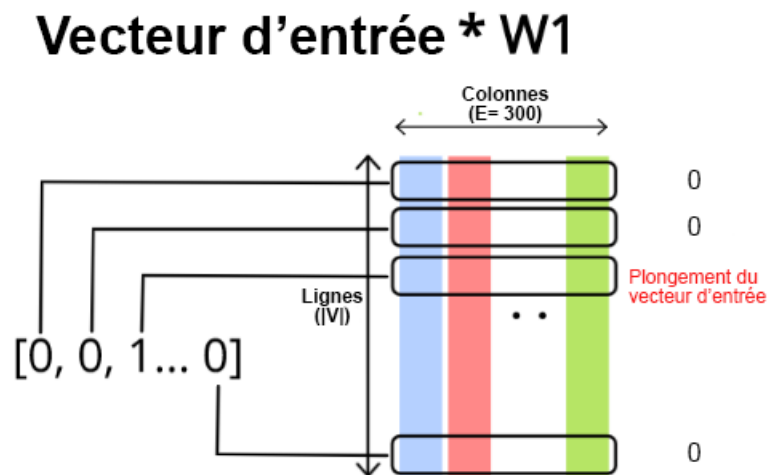


FIGURE 3.2 – Calcul d'un plongement lexical

Architecture :

L'architecture du réseau Skip-gram prend en entrée le vecteur du mot cible sous son format one-hot de dimensions $1 \times V$. Il existe une unique couche cachée. L'unique couche cachée a une dimension $V \times N$, où N est un hyperparamètre représentant la taille du plongement lexical et V la taille du vocabulaire. Le vecteur d'entrée sera multiplié à la matrice (de la couche cachée) W_1 de dimensions $N \times V$, qui retournera un vecteur $1 \times V$. Le vecteur résultant, sera ensuite multiplié à la matrice W_2 et une fonction softmax est appliqué au vecteur obtenu. La i ème composante de ce vecteur représente la probabilité que le i ème mot du vocabulaire soit un mot contexte du mot cible en entrée. La figure suivante (3.3) présente l'architecture du réseau Skip-gram :

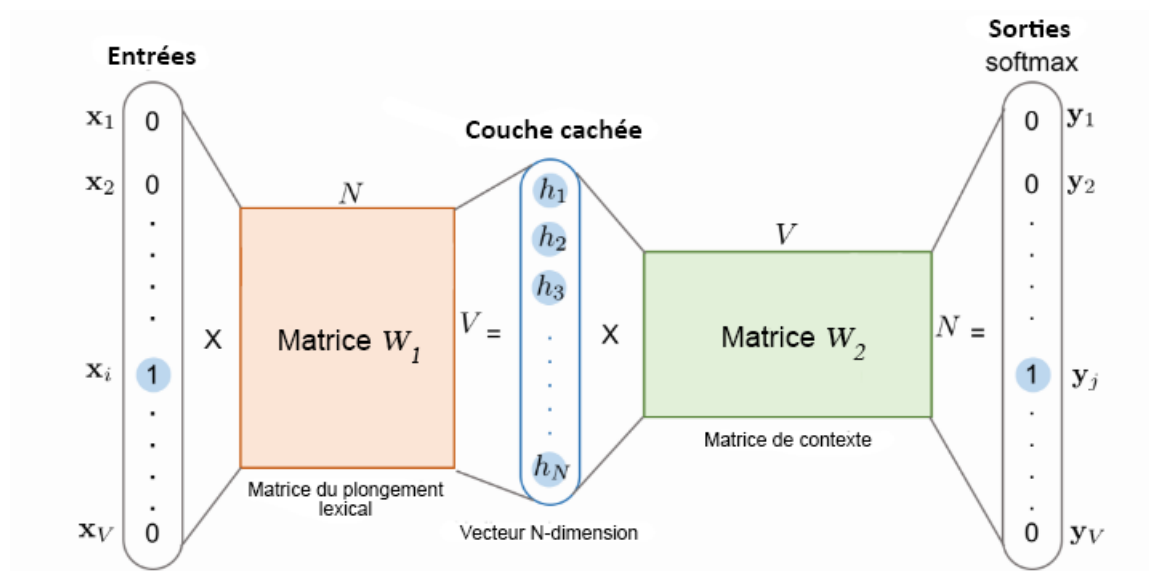


FIGURE 3.3 – Architecture du modèle Skip-gram[52]

3.6.1.b Sac de mots continu (Continuous Bag Of Words)

Le modèle sac de mots continu, (CBOW pour Continuous Bag of Words, en anglais), permet de générer la représentation d'un mot à partir de son contexte. Elle repose sur la modélisation de la probabilité conditionnelle d'observer le mot i , sachant le contexte $c = \langle w_{j1}, w_{j2}, \dots \rangle$, dans lequel il apparaît [51] :

$$p(w_i|c; U, V) = \frac{e^{u_i^\top \alpha_c}}{\sum_{i'=1}^n e^{u_{i'}^\top \alpha_c}} \quad (3.4)$$

ou :

c : Le corpus de vocabulaire de taille N .

U : La matrice représentant les mots cibles, avec pour chaque mot w_i son vecteur ligne u_i le représentant.

V : La matrice représentant les mots contextes, avec pour chaque mot w_j son vecteur ligne u_j le représentant.

α_c : Vecteur réel représentant le contexte c , représentant la moyenne des vecteurs de mots du contexte : $\frac{1}{|c|} \sum_{w_j \in c} u_j$

La fonction de vraisemblance du modèle est donc la probabilité de générer chaque mot central cible à partir de chacun des mots contextes, et est donnée par :

$$\prod_{(w_i, c) \in D} \frac{e^{u_i^\top \alpha_c}}{\sum_{i'=1}^n e^{u_{i'}^\top \alpha_c}} \quad (3.5)$$

L'objectif de cet apprentissage serait alors de maximiser la vraisemblance du jeu de données :

$$\max_{U, V} \prod_{(w_i, c) \in D} \frac{e^{u_i^\top \alpha_c}}{\sum_{i'=1}^n e^{u_{i'}^\top \alpha_c}} \quad (3.6)$$

En introduisant le logarithme :

$$\max_{U, V} \log \prod_{(w_i, c) \in D} \frac{e^{u_i^\top \alpha_c}}{\sum_{i'=1}^n e^{u_{i'}^\top \alpha_c}} = \max_{U, V} \sum_{(w_i, c) \in D} \frac{e^{u_i^\top \alpha_c}}{\sum_{i'=1}^n e^{u_{i'}^\top \alpha_c}} \quad (3.7)$$

Architecture :

Pour k nombre de mots du contexte, il y a k vecteurs d'entrée de dimensions $1 \times V$. L'unique couche cachée a une dimension $V \times N$, où N est un hyper-paramètre représentant la taille du plongement lexical. Chaque vecteur sera multiplié à la matrice de plongement lexical W_1 , qui retournera des vecteurs $1 \times V$, une moyenne sera alors calculée afin d'obtenir l'activation qui sera passée à la couche de sortie.

La i ème composante du vecteur de sortie représente la probabilité que le i ème mot du vocabulaire soit un mot cible des k mots contextes en entrée. La figure suivante (3.4) présente l'architecture du réseau CBOW :

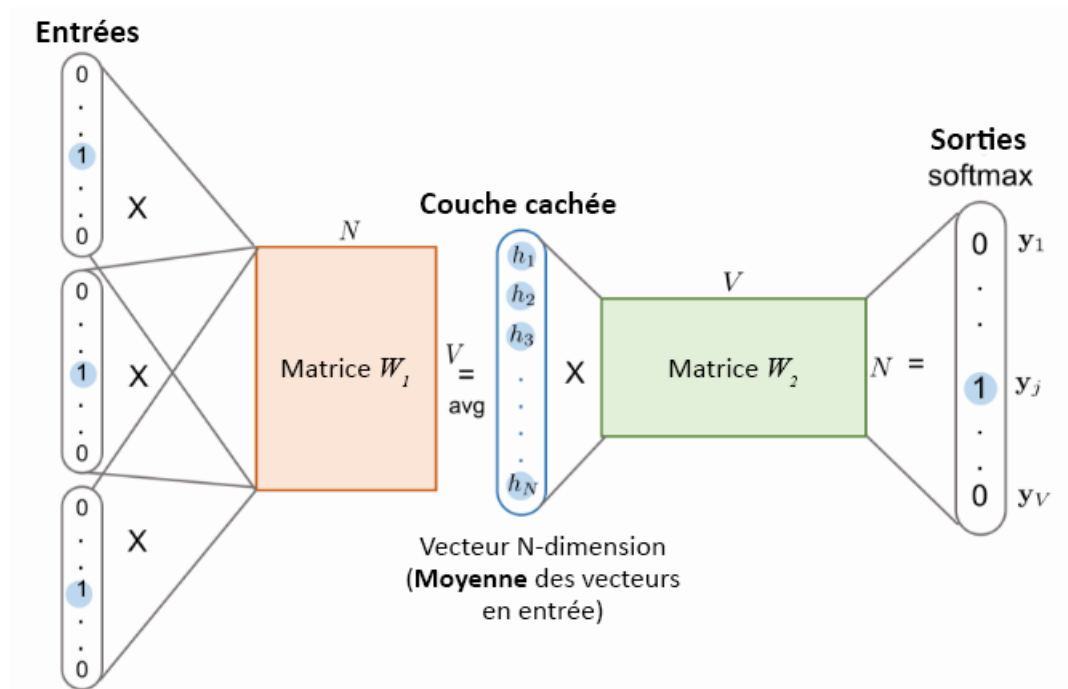


FIGURE 3.4 – Architecture du modèle CBOW[52]

Remarque :

Le calcul du plongement lexical se fera comme pour le modèle Skip-gram.

3.6.2 Vecteurs Globaux (GloVe)

Le modèle GloVe (Global Vector) a été proposé par Pennington[53] en 2014 et est basé sur l'idée que le sens des mots est mieux perçu par les ratios de cooccurrences. Cette méthode présente la génération des plongements lexicaux comme un problème de régression et modélise la fréquence de cooccurrences X_{ij} comme suit [54] :

$$u_i^\top \cdot v_j + b_i^U + b_j^V = \log(X_{ij})$$

avec :

b_i et b_j des biais, permettant d'atténuer l'erreur (par exemple une cooccurrence entre w_i ou w_j est un mot vide, qui peut être élevé, sans qu'il existe de similarité sémantique entre les mots)

X_{ij} la fréquence de cooccurrence entre les mots i et j

U : La matrice représentant les mots cibles, avec pour chaque mot w_i son vecteur ligne u_i le représentant.

V : La matrice représentant les mots contextes, avec pour chaque mot w_j son vecteur ligne v_j le représentant.

Les étapes de génération des plongements lexicaux selon GloVe sont décrites comme suit [54] :

1. Génération de la matrice de cooccurrences :

La matrice de cooccurrences³ X est générée en parcourant le corpus C

2. Définition du problème de factorisation :

Le problème de factorisation de la matrice X est définie avec une fonction d'erreur L :

$$L = \sum_{i=1}^N \sum_{j=1}^N f(X_{ij})(u_i^\top \cdot v_j + b_i^U + b_j^V - \log(X_{ij}))^2$$

L'objectif étant de minimiser cette fonction L :

$$\operatorname{argmin}_{U,V,b^U,b^V} \sum_{i=1}^N \sum_{j=1}^N f(X_{ij})(u_i^\top \cdot v_j + b_i^U + b_j^V - \log(X_{ij}))^2$$

où f est une fonction de pondération permettant de réduire l'importance des cooccurrences rares.

3. Calcul des vecteurs :

Afin de minimiser la fonction d'erreur, la méthode du gradient sera utilisée suivant les étapes ci-dessous :

- Initialiser aléatoirement les matrices U et V
- Parcourir la matrice X et modifier les vecteurs u_i et v_j de chaque couple (i, j) de X où $f(X_{ij}) > 0$ dans la direction opposée de leur gradient, les biais sont aussi modifiés dans la direction opposée de leurs dérivées partielles

3. Une matrice de cooccurrences est une matrice de comptage du nombre d'apparitions des mots lignes (mot central) dans le contexte de fenêtre n des mots colonnes.

3. Une fenêtre de contexte est le nombre de mots à considérer comme appartenant au contexte. Autrement dit, les mots présents à droite et à gauche du mot central (mot ligne dans ce cas).

3.6.3 FastText

La méthode FastText a été introduite par Facebook en 2016. Il s'agit d'une extension des modèles Word2Vec, la différence réside dans la représentation des mots. FastText représente chaque mot comme un n-grams de ses caractères, permettant de mieux représenter les mots courts ainsi que les affixes.

Exemple :

Le mot *artificial* sera représenté $\langle ar, art, rti, tif, ifi, fic, ici, ial, al \rangle$ pour $n = 3$

Un modèle Skip-gram est ensuite entraîné sur les n-grams obtenus afin d'apprendre les plongements lexicaux.

FastText présente un avantage majeur, en comparaison avec les autres méthodes, étant la possibilité de représenter des mots qui ne sont pas dans le vocabulaire utilisé lors de l'entraînement du modèle.

3.7 Apprentissage Automatique

L'apprentissage automatique fait référence au développement, à l'analyse et à l'implémentation de méthodes qui permettent à une machine (au sens large) d'évoluer grâce à un processus d'apprentissage, et ainsi de remplir des tâches qu'il est difficile ou impossible de remplir par des moyens algorithmiques plus classiques.[55]

Il existe de nombreux types d'apprentissage : L'apprentissage supervisé, l'apprentissage non-supervisé, l'apprentissage par renforcement et l'apprentissage semi-supervisé. Nous nous intéressons seulement à l'apprentissage supervisé, non supervisé et semi-supervisé.

3.7.1 Apprentissage Supervisé

Definition 3.7.1 *Ensemble d'apprentissage :*

Un ensemble d'apprentissage est un ensemble de N couples d'entrée-sortie $(x_i, y_i) : x_i \in X$ et $y_i \in Y$, $1 \leq i \leq N$. On appelle x_i le vecteur des attributs et y_i la classe.

Definition 3.7.2 *Apprentissage supervisé :*

Soit D un ensemble fini d'apprentissage de N éléments $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, On appelle apprentissage supervisé, le processus permettant de déterminer la fonction f tel que[56] :

$$\begin{aligned} f : X &\rightarrow Y \\ x &\mapsto \hat{y} = f(x) \end{aligned} \tag{3.8}$$

$Y \subset R$: lorsque la valeur à estimer est dans un ensemble infini et continu, il s'agit d'un problème de régression.

$Y \subset \{1, \dots, m\}$: lorsque l'ensemble de valeurs de la sortie est fini et discret, il s'agit d'un problème de classification qui revient à attribuer une étiquette à chaque entrée.

3.7.2 Apprentissage Non Supervisé

Definition 3.7.3 *Partitionnement :*

Soit $Y = y_1, y_2, \dots, y_N$ un ensemble de N éléments, tel que $y_i \in R^n$

Si $c \in N$ tel que $2 \leq c \leq n$, on appelle c -partition de Y le tuple (Y_1, Y_2, \dots, Y_c) des sous-ensembles de Y satisfaisant les conditions suivantes :

$$Y_i = \emptyset \quad 1 \leq i \leq c \tag{3.9}$$

$$\bigcup_{i=1}^c Y_i = Y \tag{3.10}$$

$$Y_i \cap Y_j = \emptyset \quad i \neq j \tag{3.11}$$

Definition 3.7.4 *Clustering :*

Soit $Y = \{y_1, y_2, \dots, y_n\}$ un ensemble de N éléments, tel que $y_i \in R^n$, un clustering de Y consiste en un partitionnement de Y vérifiant que la similarité intra-cluster est maximale et la similarité inter-cluster est minimale. Ces deux critères représentent respectivement la similarité entre les éléments d'une même partition et la similarité entre les éléments de partitions différentes.

3.7.3 Apprentissage Semi-Supervisé

Le problème d'apprentissage semi-supervisé peut être classé avec les problèmes d'apprentissage supervisé, l'objectif étant de déterminer la fonction satisfaisant 3.8. La différence entre ce problème et un problème de classification standard réside dans la présence de données non labellisées $D_u = \{x_{n+j} | j = 1, \dots, m\}$ en plus des données [57] labellisées $D_l = \{(x_i, y_i) | i = 1, \dots, n\}$.

Il existe deux principales approches de l'apprentissage semi-supervisé [57] :

1. Considérer le problème comme exclusivement supervisé en ignorant D_u
2. Traiter y comme une variable de classe latente⁴ et estimer les groupes latents à l'aide d'une méthode non supervisée, puis associer chaque groupe latent avec les classes observées en utilisant D_l .

L'étiquetage des données étant une tâche coûteuse et difficile, l'apprentissage semi-supervisé peut alors être une solution intéressante lorsque la quantité de données non labellisées est très grande et celles labellisées moins importantes.

3.7.4 Méthodes d'apprentissage**3.7.4.a Méthodes d'apprentissage supervisé**

Il existe différentes méthodes d'apprentissage supervisé :

1. K-Plus Proches Voisins (KNN) :

Soit A un ensemble d'apprentissage, on souhaite associer à une entrée x non étiquetée une classe. La méthode KNN s'exécute comme suit :

- Trouver les k éléments les plus proches d'une entrée x en utilisant une mesure de similarité.
- Associer à x la classe majoritaire parmi les k -plus proches éléments.

4. Variable non directement observable mais dont les valeurs peuvent être estimées à partir de données observables.

La figure suivante (3.5) présente une exécution de l'algorithme KNN et la classification d'une entrée x pour $k = 6$ et $k = 3$:

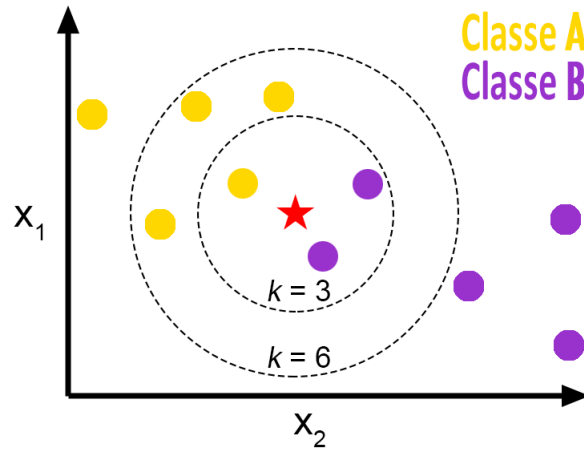


FIGURE 3.5 – Pour $k=6$, l'entrée (en rouge) sera affecté à la classe A

2. Machine à vecteur de support (SVM) :

Definition 3.7.5 Hyperplan :

Soit E un espace vectoriel à n dimensions, on appelle hyperplan de E tous les sous espaces de taille $n-1$ qui sont dans E .

La machine à vecteur de support est une méthode de classification qui consiste à rechercher l'hyperplan linéaire de séparation optimal. La classification d'un élément x se fait selon la position de x par rapport à l'hyperplan, comme le montre la figure suivante (3.6) :

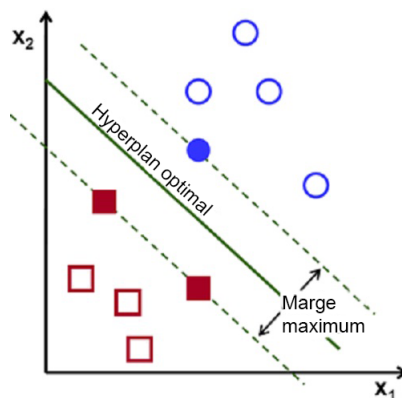


FIGURE 3.6 – L'hyperplan optimal maximise la marge, les points sur les lignes en pointillés sont appelés supports.

3. Classifieur Bayésien Naïve :

Cette méthode est basée sur le théorème de Bayes :

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \quad (3.12)$$

Soit un ensemble de paramètres $X = (x_1, x_2, \dots, x_n)$ et une classe y à déterminer. On supposera l'indépendance entre les variables. En remplaçant X dans 3.12 et en simplifiant :

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(x_1|y)P(x_2|y)\dots P(x_n|y)P(y)}{P(x_1)P(x_2)\dots P(x_n)} \quad (3.13)$$

Le dénominateur $P(x_1)P(x_2)\dots P(x_n)$ étant constant pour le dataset donné, on peut alors écrire :

$$P(y|x_1, x_2, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i|y) \quad (3.14)$$

L'objectif est alors de déterminer la classe y maximisant 3.14

$$y = \operatorname{argmax}_y P(y) \prod_{i=1}^n P(x_i|y) \quad (3.15)$$

3.7.4.b Méthodes d'apprentissage non supervisé

1. K-Moyennes

Cet algorithme attribue de façon répétée les points de données au centroïde le plus proche en utilisant une distance moyenne entre le centroïde et les points de données. Les centroïdes sont actualisés en calculant le barycentre⁵ des points de chaque classe. L'ensemble de données est divisé en classes ou partitions appelées clusters.

2. Clustering hiérarchique :

Le clustering hiérarchique consiste à grouper les objets en une hiérarchie. On distingue deux types de clustering hiérarchique :

(a) Agglomératif :

- Chaque point est considéré comme un cluster, puis les deux clusters les plus proches sont regroupés (agglomérés) en un seul cluster.
- La procédure est réitérée jusqu'à l'obtention d'un cluster regroupant l'ensemble des clusters initiaux.

(b) Divisif :

- Initialiser un cluster contenant tous les points.
- Séparer itérativement chaque cluster en plusieurs, en regroupant les points les plus similaires, jusqu'à ce que chaque point représente son propre cluster.

3.7.4.c Méthodes d'apprentissage semi-supervisé

1. Auto apprentissage (Self-training)

Cette technique peut être appliquée avec toutes les méthodes d'apprentissage supervisé.

Soit $D_l = \{(x_i, y_i) | i = 1, \dots, n\}$ l'ensemble des données labellisées et $D_u = \{x_n + j | j = 1, \dots, m\}$ l'ensemble des données non labellisés.

Les étapes sont décrites ci-dessous :

- (a) Entraîner f sur l'ensemble D_l à l'aide d'un apprentissage supervisé.
- (b) Classifier les données de l'ensemble D_u
- (c) Retirer un sous-ensemble R à l'ensemble D_u .
- (d) Ajouter $\{x, f(x) | x \in R\}$ à l'ensemble D_l
- (e) Répéter jusqu'à ce que $D_u = \emptyset$

5. centre de gravité

2. Estimation Maximisation

Cette méthode admet l'hypothèse de distribution des données en k classes, chacune suivant une loi normale. Il s'agit d'une méthode de clustering flou c'est à dire où l'appartenance à une classe est calculée à l'aide de probabilités. L'algorithme est utilisé en classification semi-supervisée et non supervisé afin de déterminer la moyenne ainsi que la variance de chaque distribution. Il se présente en deux étapes [58] :

- (a) Calcul de l'espérance de la vraisemblance à l'aide de la variance σ^2 et de la moyenne μ .
- (b) Maximisation de la vraisemblance à l'aide des paramètres calculés à l'étape précédente.

Lorsqu'elle est utilisée pour un apprentissage supervisé, l'espérance de la vraisemblance est mise à 1 pour les points appartenant à l'ensemble étiqueté.

La figure suivante 3.7 représente les itérations de l'algorithme EM :

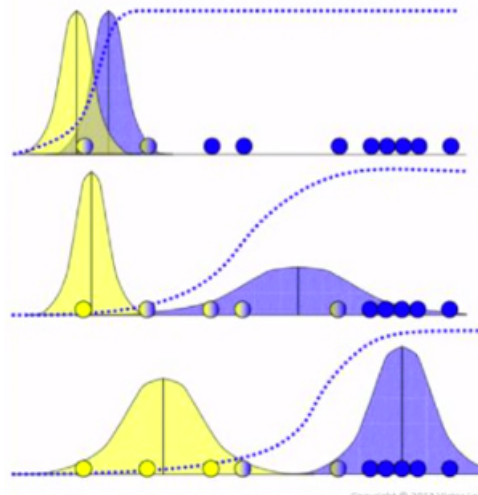


FIGURE 3.7 – Exemple d'itérations de l'algorithme EM

3.8 Réseaux de Neurones Artificiels

3.8.0.a Perceptron

Le perceptron est un classifieur linéaire et binaire, utilisé en apprentissage supervisé. Il s'agit du type de réseaux de neurones artificiels le plus simplifié. Un perceptron[59] est défini par :

$$y = f\left(\sum_{i=1}^N w_i x_i - \theta\right) = \begin{cases} 1 & \text{si } \sum_{i=1}^N w_i x_i - \theta > 0 \\ 0 & \text{sinon} \end{cases} \quad (3.16)$$

où :

w_i sont appelés coefficients synaptiques ou poids

N est le nombre d'entrées

f est appelée fonction d'activation

θ est appelé biais (ou seuil)

Le modèle peut être simplifié en ajoutant une entrée supplémentaire $x_0 = 1$ et un poids $w_0 = -\theta$, donnant ainsi :

$$y = \begin{cases} 1 & \text{si } \sum_{i=0}^N w_i x_i > 0 \\ 0 & \text{sinon} \end{cases} \quad (3.17)$$

La figure suivante présente un perceptron à n entrées et $n + 1$ poids :

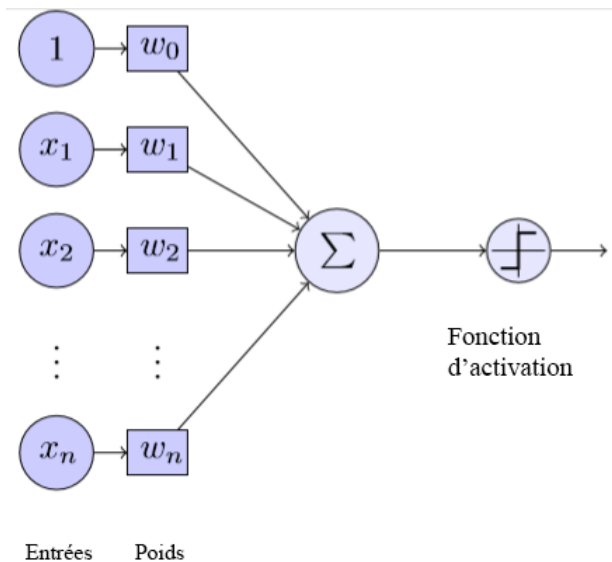


FIGURE 3.8 – Perceptron

Calcul de l'erreur

Soit un ensemble d'apprentissage de N exemples, (x_k, c_k) le k -ième élément de l'ensemble et y_k la sortie retournée par le réseau. L'erreur est alors définie par [60] :

$$E_k = (y_k - c_k) \quad (3.18)$$

3.8.0.b Algorithme d'apprentissage par correction d'erreur

L'algorithme suivant permet de déterminer les poids d'un perceptron dans le but de classifier correctement en ensemble d'apprentissage.

Algorithm 1 Algorithme d'apprentissage par correction d'erreur [59]

- 1: D : Ensemble d'apprentissage $(\{0, 1\}^n \times \{0, 1\}$ ou $R^n \times \{0, 1\})$
 - 2: **Pour** $i = 0..n$ initialiser aléatoirement les poids w_i
 - 3: **Repeter**
 - 4: Sélectionner un couple (\vec{x}, c) dans D
 - 5: Calculer o la sortie pour l'entrée \vec{x}
 - 6: **Pour** $i = 0..n$
 - 7: $w_i = w_i + (c - o) * x_i$
 - 8: **Fin Pour**
 - 9: **Fin Répéter**
-

3.8.0.c Algorithme d'apprentissage par descente du gradient

La descente du gradient est une méthode d'optimisation permettant de trouver le minimum d'une fonction convexe par convergence. On définit la fonction d'erreur d'un perceptron sur l'ensemble d'exemples D par :

$$E(\vec{w}) = 1/2 \sum_{(\vec{x}, c) \in D} (c - o)^2 \quad (3.19)$$

avec :

o : sortie retournée par le perceptron pour une entrée \vec{x}

Algorithm 2 Algorithme d'apprentissage par descente du gradient [59]

```

1:  $D$  : Ensemble d'apprentissage ( $\{0, 1\}^n \times \{0, 1\}$  ou  $R^n \times \{0, 1\}$ )
2:  $\epsilon$  : paramètre empirique
3: Pour  $i = 0..n$  initialiser aléatoirement les poids  $w_i$ 
4: Répéter
5:   Pour tout  $i$  :  $\Delta w_i = 0$ 
6:   Pour tout couple  $(\vec{x}, c)$  de  $D$ 
7:     Calculer la sortie  $o$ 
8:     Pour tout  $i$  :  $\Delta w_i = \Delta w_i + \epsilon(c - o)x_i$  avec  $\frac{\partial E}{\partial w_i} = \epsilon(c - o)$ 
9:   Fin Pour
10:  Pour tout  $i$  :  $w_i = w_i + \Delta w_i$ 
11: Jusqu'à satisfaction de la condition d'arrêt
12: Fin

```

Il existe différentes stratégies de descente du gradient :

- **Descente de Gradient classique** : Les gradients ne sont calculés, et les poids mis à jour qu'après le passage complet de l'ensemble d'apprentissage.
- **Descente de Gradient en mode incrémental (stochastique)** : Les gradients sont calculés et les poids mis à jour après chaque passage d'exemple de l'ensemble d'apprentissage.
- **Descente de Gradient par traitement par lots (mini batch)** : Les gradients sont calculés et les poids mis à jour après le passage d'un lot d'exemples. La taille des lots est un paramètre empirique (hyper-paramètre).

3.8.0.d Perceptron multicouches

Le perceptron simple est limité aux problèmes linéairement séparable⁶. Afin de palier à cette limitation, le perceptron multicouche est introduit. On appelle couche un ensemble de perceptrons. Un perceptron multicouche est composé d'une couche d'entrée, une ou plusieurs couches cachées et d'une couche de sortie, comme le montre la figure 3.9 Les perceptrons sont présents dans la ou les couches cachées et la couche de sortie. Leurs fonctions d'activation peuvent être linéaires ou non-linéaires.

6. Un échantillon $S = (x_1, y_1), \dots, (x_l, y_l)$ est linéairement séparable s'il existe un classifieur linéaire qui classe correctement tous les exemples de S .

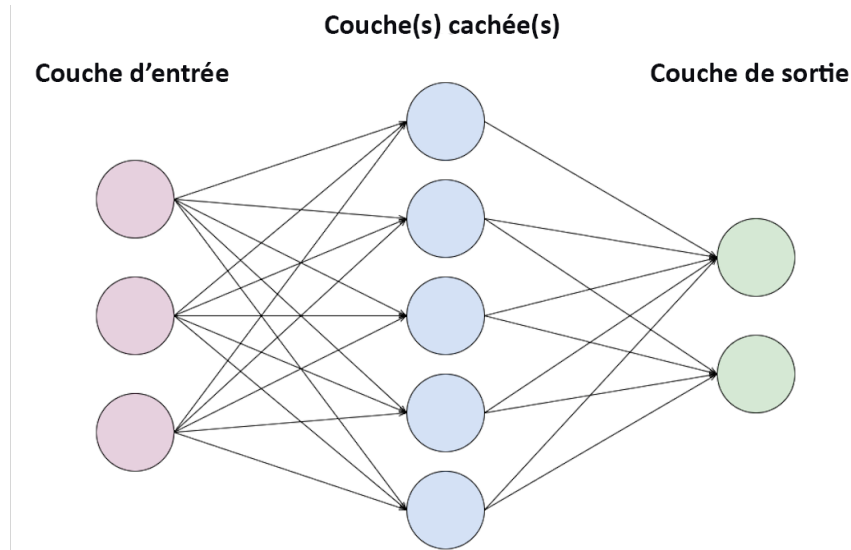


FIGURE 3.9 – Perceptron multicouche

3.8.0.e Types de fonctions d'activations

Il existe deux familles de fonctions d'activation :

1. Fonctions Linéaires :

- Fonction de Heaviside (défini en 3.16)
- Fonction Identité : $f(x) = ax$

2. Fonctions non-linéaires :

- Fonction Sigmoid : $f(x) = \frac{1}{1+e^{-x}}$
- Fonction Tangente hyperbolique : $f(x) = \tanh(x)$

- Fonction Softmax : $\sigma(x)_j = \frac{e^{x_j}}{\sum_{k=1}^K e^{x_k}}$

$\sigma(x)_j$ la valeur de la composante j du vecteur $\sigma(x)$.

(Cette fonction prend en entrée un vecteur $x = \{x_1, x_2, \dots, x_K\}$ de K éléments)

- Fonction ReLu : $f(x) = \begin{cases} 0 & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$

3.8.0.f Fonctions d'erreur :

Definition 3.8.1 Une fonction d'erreur est utilisée afin de mesurer la performance de prédiction d'un modèle, c'est à dire sa capacité à prédire le résultat attendu. La valeur de l'erreur calculée est nulle si la prédiction du modèle est parfaite.

Similairement, les fonctions d'erreur peuvent être classifiées en deux catégories :

1. Fonctions d'erreur de classification :

— Entropie croisée (Log loss)

permet de mesurer la performance d'un modèle de classification dont la sortie est une probabilité $[0 - 1]$. Soit N le nombre de classe totale, $P_{o,c}$ la probabilité que l'exemple o appartienne à la classe c prédite par le modèle, et $y_{o,c}$ valeur binaire d'appartenance ou non de l'exemple o à la classe c .

Pour une classification multiclasse ($N > 2$) :

$$L = - \sum_{c=1}^N y_{o,c} \log(P_{o,c})$$

Pour une classification binaire ($N = 2$) :

$$L = -y \log p + (1 - y) \log (1 - p)$$

— Entropie Relative (Kullback-Leibler Divergence)

Cette fonction permet de mesurer la différence entre deux distributions de probabilités $p(x)$ et $q(x)$ sur une variable x . Autrement dit, l'entropie relative $p(x)$ par rapport à $q(x)$, notée $D_{KL}(p||q)$ mesure l'information perdue lorsque q est utilisé comme une approximation de p .

$$D_{KL}(p||q) = \sum_{i=1}^N p(x_i) (\log p(x_i) - \log q(x_i))$$

2. Fonctions d'erreur de régression

— Erreur moyenne absolue

Définie par la formule suivante :

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

— Erreur moyenne quadratique

Définie comme suit :

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

3.8.0.g Algorithme de Rétropropagation (Backpropagation)

Cet algorithme permet d'entraîner les réseaux de neurones multicouches. Le but étant de minimiser la fonction d'erreur en modifiant les poids et ce en propageant l'information sur la dérivée de la fonction d'erreur de la couche de sortie jusqu'à la couche d'entrée.

Algorithm 3 Algorithme de rétro-propagation [61]

- 1: D : Ensemble d'apprentissage
 - 2: r : paramètre empirique appelé taux d'apprentissage
 - 3: Initialiser tous les poids du réseau w_{ij} .
 - 4: **Répéter :**
 - 5: **Pour chaque** exemple (x_i, y_i) de D faire :
 - 6: Propager l'exemple $v_i = \sum w_{ij}x_j$ et $\hat{y}_i = f(v_i)$ (f la fonction d'activation)
 - 7: **Pour chaque** neurone de sortie i faire :
 - 8: $\delta_i = -f'(v_i)(y_i - \hat{y}_i)$
 - 9: **Pour chaque** neurone caché i faire :
 - 10: $\delta_i = f'(v_i) \sum_{k \in \text{outputs}} w_{ki} \delta_k$
 - 11: Mettre à jour les poids :
 - 12: $w_{ij} := w_{ij} - r \frac{\partial E}{\partial w_{ij}}$ avec $\frac{\partial E}{\partial w_{ij}} = \delta_i a_j$
 - 13: a_j la sortie d'un neurone caché ou de sortie
 - 14: **Jusqu'à satisfaction des critères d'arrêt**
 - 15: Retourner le réseau
-

3.8.1 Réseaux de Neurones Récurrents (RNN)

Les réseaux de neurones multicouches décrits précédemment admettent l'indépendance entre les inputs. Cette hypothèse est préjudiciable lorsque les données en entrée sont séquentielles et que chacune d'elle dépend de ce qui la précède. La figure suivante présente l'architecture d'un réseau de neurones récurrent :

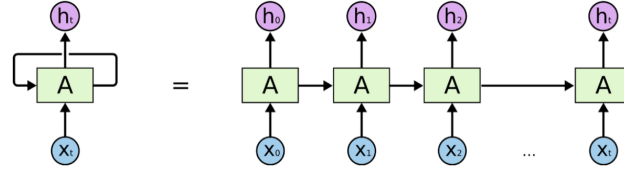


FIGURE 3.10 – Architecture d'un réseau de neurones récurrents

Les réseaux de neurones récurrents sont des classes de réseaux de neurones qui permettent de prendre en entrée des données séquentielles. Ils sont dotés d'une mémoire, de telle sorte qu'à chaque étape t , chaque couche cachée h_t recevra en entrée l'input courant x_t ainsi que l'activation $a^{<t-1>}$ de la couche cachée précédente [62]. Comme le montre la figure 3.11, l'activation $a^{<t>}$ et la sortie $y^{<t>}$ à une étape t seront définies par les formules suivantes :

$$a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^t + b_a) \quad (3.20)$$

$$y^{<t>} = g_2(W_{ya}a^{<t>} + b_y) \quad (3.21)$$

où :

$W_{aa}, W_{ax}, W_{ya}, b_a, b_y$ sont des paramètres et g_1, g_2 des fonctions d'activation.

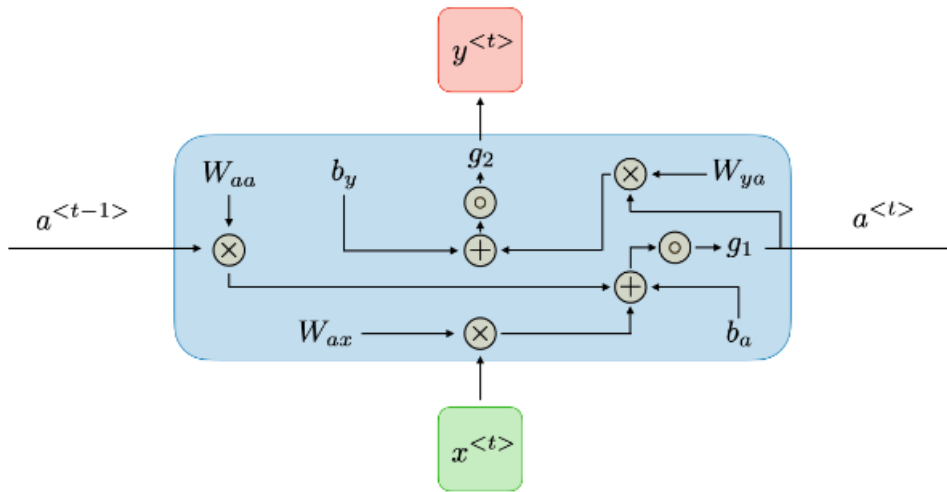


FIGURE 3.11 – Architecture d'une cellule RNN[62]

3.8.1.a Types de réseaux récurrents

Il existe de nombreux types de réseaux récurrents :

- One-to-one : $T_x = 1$ et $T_y = 1$
- One-to-many : $T_x = 1$ et $T_y > 1$
- Many-to-one : $T_x > 1$ et $T_y = 1$
- Many-to-many : $T_x > 1$ et $T_y > 1$

avec :

T_x le nombre d'inputs et T_y le nombre d'outputs du réseau.

En pratique, le choix du type se fait selon les besoins de l'application.

3.8.1.b Fonction d'erreur

La fonction d'erreur d'un réseau récurrent est la somme des erreurs à chaque étape t

$$L(\hat{y}, y) = \sum_{t=1}^{T_y} L(\hat{y}^{<t>}, y^{<t>}) \quad (3.22)$$

3.8.1.c Rétropropagation dans le temps

La rétropropagation dans le temps (Backpropagation through Time) est une adaptation de l'algorithme de rétropropagation aux réseaux récurrents. A chaque étape t , la dérivée partielle de l'erreur L par rapport à la matrice des poids W est exprimée par :

$$\frac{\partial \mathcal{L}^{(T)}}{\partial W} = \sum_{t=1}^T \frac{\partial \mathcal{L}^{(t)}}{\partial W} \Big|_{(t)} \quad (3.23)$$

3.8.1.d Disparition du gradient

Les réseaux récurrents deviennent de plus en plus limités lorsque le nombre de couches cachées N augmente. Ceci est dû au problème de disparition du gradient. Ce problème est provoqué par le caractère multiplicatif des gradients. Ces derniers décroissent exponentiellement lorsque le nombre de couches augmente. Il en résulte une modification des poids minime notamment pour les premières couches et une mauvaise convergence vers la solution. Ce phénomène explique pourquoi les réseaux récurrents capturent difficilement les dépendances à long terme dans les séquences.

3.8.2 Réseaux Récurrents à Mémoire Court et Long terme (LSTM)

Le réseau LSTM (pour Long short-term memory) est une extension des réseaux récurrents classiques. Ces réseaux ont une structure similaire aux réseaux récurrents.

Comme sur la figure 3.12, chaque couche (ou cellule) du réseau possède un état caché $h^{(t)}$ et état de cellule $c^{(t)}$. Il s'agit de vecteurs de taille n .

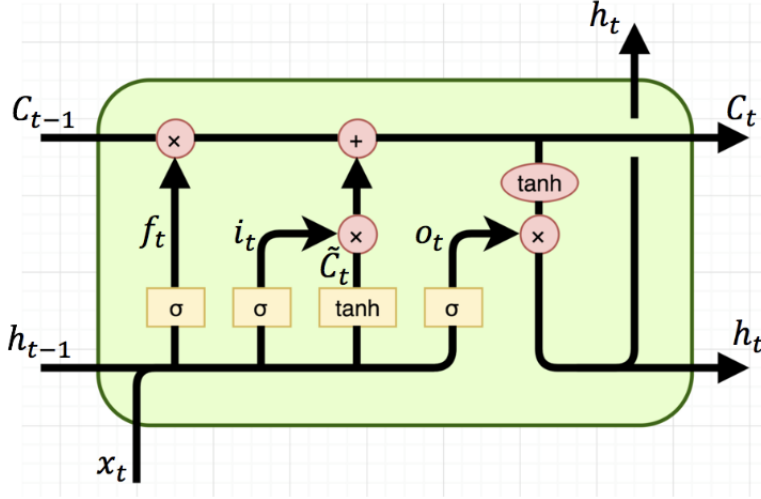


FIGURE 3.12 – Architecture interne d'une cellule LSTM

Architecture d'une cellule LSTM

Chaque cellule LSTM possède [63] :

1. **Porte de mise à jour** (input gate) : permet d'autoriser ou bloquer la mise à jour de l'état $c^{(t)}$ et est définie par :

$$i^{(t)} = \sigma(W_i h^{(t-1)} + U_i x^{(t)} + b_i)$$

2. **Porte de sortie** (Output gate) : Contrôle la communication de l'état de la cellule en sortie et est définie par :

$$o^{(t)} = \sigma(W_o h^{(t-1)} + U_o x^{(t)} + b_o)$$

3. **Porte de remise à zéro** (forget gate) : Contrôle si l'information arrivant de la cellule précédente sera gardée ou pas et est définie par :

$$f^{(t)} = \sigma(W_f h^{(t-1)} + U_f x^{(t)} + b_f)$$

4. **Nouveau contenu de la cellule** : contenu produit par la cellule et est défini par :

$$\tilde{c}^{(t)} = \tanh(W_c h^{(t-1)} + U_c x^{(t)} + b_c)$$

5. **État de la cellule** : Supprime une partie du contenu de la cellule précédente et ajoute du contenu nouveau et est défini par :

$$c^{(t)} = f^{(t)} * c^{(t-1)} + i^{(t)} * \tilde{c}^{(t)}$$

6. **État caché** : retourne une partie du contenu de la cellule et est défini par :

$$c^{(t)} = o^{(t)} * \tanh c^{(t)}$$

σ est la fonction sigmoïde.

3.8.3 Réseaux de Neurones Récurrents à Portes (GRU)

Les réseaux GRU (Gated Recurrent Units) représentent une alternative plus simple aux réseaux LSTM. A chaque étape t , une cellule possède un input $x^{(t)}$ et un état caché $h^{(t)}$, comme sur la figure 3.13. Ces réseaux ont la particularité d'avoir des calculs moins longs lors de l'apprentissage et moins de paramètres que les réseaux LSTM.

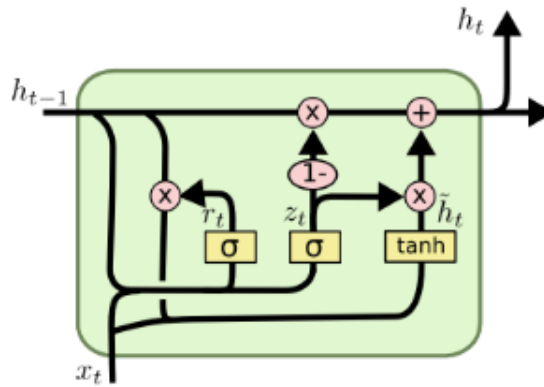


FIGURE 3.13 – Architecture interne d'une cellule GRU

Architecture d'une cellule GRU [63]

1. **Porte de mise à jour** (Update gate) : contrôle quelles parties de l'état caché seront mises à jour ou conservées.

$$u^{(t)} = \sigma(W_u h^{(t-1)} + U_u x^{(t)} + b_u)$$

2. **Porte de réinitialisation** (Reset gate) : détermine quelles parties de l'état caché précèdent $h^{(t-1)}$ seront utilisées pour calculer le nouveau contenu de la cellule.

$$r^{(t)} = \sigma(W_r h^{(t-1)} + U_r x^{(t)} + b_r)$$

3. **Nouveau contenu de l'état caché** : Calcule le nouveau contenu de l'état caché.

$$\tilde{h}^{(t)} = \tanh(W_h(r^{(t)} * h^{(t-1)} + U_h x^{(t)} + b_h))$$

4. **État caché** : Contenu qui sera passé à la cellule de l'étape $t + 1$

$$h^{(t)} = (1 - u^{(t)}) * h^{(t-1)} + u^{(t)} * \tilde{h}^{(t)}$$

Remarque :

Les réseaux LSTM et GRU permettent de prévenir le phénomène de disparition des gradients, et ce à l'aide de la porte de mise à zéro (forget gate) qui permet un meilleur contrôle des valeurs des gradients à chaque étape.

3.8.4 Réseaux Récurrents Bi-directionnels (Bi-RNN)

L'architecture des réseaux récurrents bidirectionnels consiste à superposer deux couches cachées indépendantes de réseaux récurrents ayant des directions opposées mais étant connectées aux mêmes entrées. La figure suivante présente l'architecture générale d'un réseau récurrent bidirectionnel.

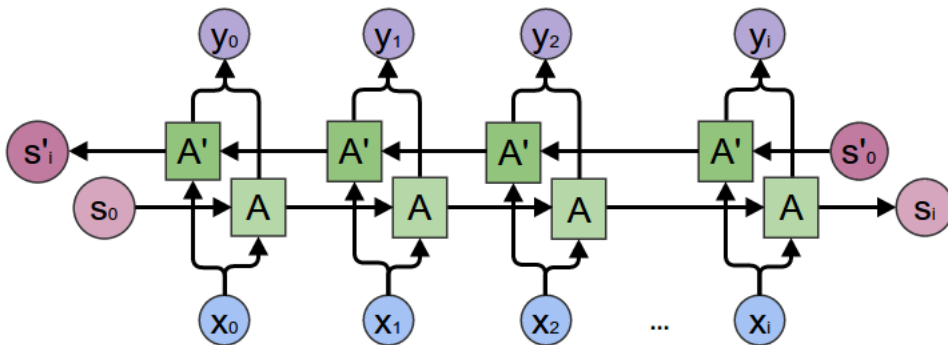


FIGURE 3.14 – Architecture générale d'un réseau récurrent bidirectionnel

Cette architecture permet à chaque étape t , de recevoir des informations des étapes précédentes comme pour les réseaux récurrents classiques, mais aussi des informations des étapes futures.

3.8.5 Réseaux de Neurones Convolutifs (CNN)

Definition 3.8.2 *Convolution* :

La convolution est une opération de sous-échantillonnage qui consiste à effectuer une somme des voisins immédiats de chaque élément de la matrice, en pondérant par les éléments d'un filtre f appelée masque de convolution.

La figure suivante montre un exemple d'une opération de pooling :

7	2	3	3	8
4	5	3	8	4
3	3	2	8	4
2	8	7	2	7
5	4	4	5	4

*

1	0	-1
1	0	-1
1	0	-1

=

6		

$$\begin{aligned}
 &7 \times 1 + 4 \times 1 + 3 \times 1 + \\
 &2 \times 0 + 5 \times 0 + 3 \times 0 + \\
 &3 \times -1 + 3 \times -1 + 2 \times -1 \\
 &= 6
 \end{aligned}$$

FIGURE 3.15 – Opérations de convolution

Definition 3.8.3 *Pooling* :

L'opération de pooling permet de réduire la dimension d'une matrice (compression) tout en conservant ses caractéristiques pertinentes. La matrice est découpée en sous matrices sur lesquelles une opération est effectuée (maximum, moyenne, etc.) et a pour paramètre le stride qui correspond au pas entre chaque sous matrice.

La figure suivante présente des exemples d'opérations de pooling max et moyenne :

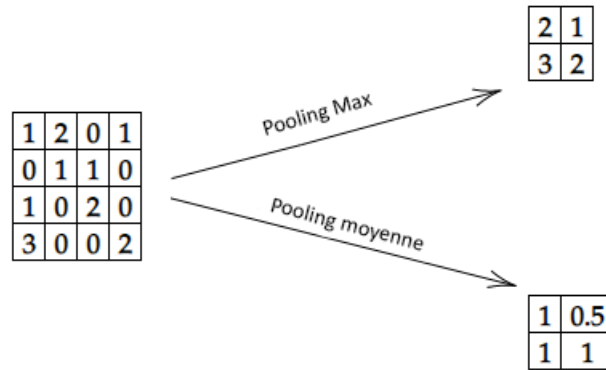


FIGURE 3.16 – Opération de pooling

Les réseaux convolutifs sont des types de réseaux particuliers, principalement utilisés sur les données visuelles (images, vidéos..). L'architecture de ces réseaux est composée des couches suivantes[64] :

1. **Couche convolutionnelle** : Une ou plusieurs opérations de convolution sont effectuées sur les données d'entrée D , à l'aide d'un masque f selon un stride S , dont les dimensions sont des hyper-paramètres.
Les données de sortie résultantes sont appelées activation map.
2. **Couche de Pooling** : L'activation map est sous-échantillonnée à l'aide de l'opération de pooling.
3. **Couche fully Connected** : chaque sortie de la couche de pooling sera ensuite connectée à l'ensemble des neurones de la couche fully connected.

3.9 Attaques Adversaires

Definition 3.9.1 *Les exemples adversaires sont des entrées de modèles d'apprentissage intentionnellement conçus afin de piéger les modèles c'est à dire en provoquant des prédictions erronées [65].*

Definition 3.9.2 *Le processus de génération des exemples adversaires est appelé attaque adversaire [65].*

De nombreuses recherches, plus particulièrement dans le domaine de la classification d'images [66], montrent que les modèles d'apprentissage profond sont vulnérables à de très petites perturbations. Ces modèles vont très probablement mal classer les exemples adversaires avec un très haut degré de confiance. Dans [14], Ian Goodfellow, illustre comment une image de panda (figure 3.17) pourrait être classée comme un singe avec un degré de confiance de 99%, et cela en ajoutant une petite perturbation, non visible à l'oeil nu, à l'image de départ.

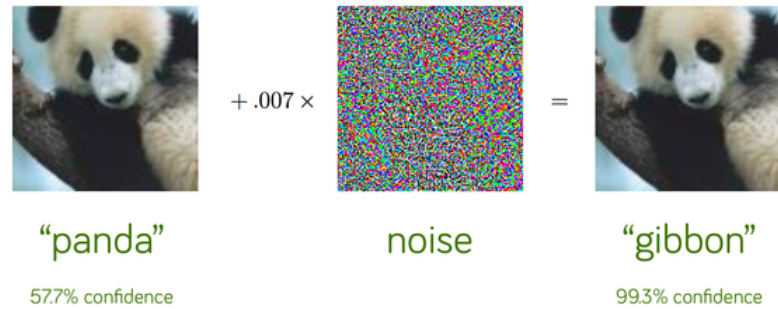


FIGURE 3.17 – Exemple de génération d'exemple adversaire

3.9.1 Types d'attaques adversaires

On dénombre deux types d'attaques adversaires :

- Attaques boîte blanche : Dans ce type d'attaque, l'attaquant a en sa possession l'ensemble des informations sur le modèle ciblé (architecture, paramètres, gradients, etc.)
- Attaques boîte noire : L'assaillant n'a pas accès aux configurations du modèle cible, il peut seulement passer des inputs au modèle et obtenir leurs outputs.

Nous nous intéressons seulement aux attaques de type boîte blanche.

Soit un classifieur h_θ de paramètres θ . La minimisation de la fonction de d'erreur ℓ sur un ensemble d'apprentissage $x_i \in X, y_i \in Y | i = 1..m$ est un problème d'optimisation décrit par :

$$\underset{\theta}{\text{minimize}} \frac{1}{m} \sum_{i=1}^m \ell(h_\theta(x_i), y_i)$$

Dans le cas d'un réseau de neurones, ce problème est résolu en utilisant la descente du gradient sur un batch B de taille m , en calculant le gradient de l'erreur par rapport aux paramètres θ .

$$\theta = \theta - \frac{\alpha}{|B|} \sum_{i \in B} \nabla_{\theta} \ell(h_{\theta}(x_i), y_i)$$

Le gradient de ℓ dans l'équation ci-dessus permet de déterminer les changements sur l'erreur lorsque les paramètres θ subissent des ajustements minimes. Similairement, déterminer un exemple adversaire revient à calculer le gradient de la fonction d'erreur par rapport à une entrée x_i . L'objectif étant de générer un exemple \hat{x} qui sera prédit incorrectement. Cela revient à modifier l'entrée afin de maximiser l'erreur :

$$\underset{\hat{x}}{\text{maximize}} \ell(h_{\theta}(\hat{x}), y)$$

Néanmoins, un exemple adversaire doit être proche de l'entrée non modifiée x , on définit la perturbation δ et Δ l'ensemble de perturbations admissibles :

$$\underset{\delta \in \Delta}{\text{maximize}} \ell(h_{\theta}(x + \delta), y)$$

3.9.2 Risques adversaires

Definition 3.9.3 *On appelle risque adversaire la quantité qui décrit l'erreur d'un classifieur au pire cas, dans le cas où chaque entrée de l'ensemble d'apprentissage peut être perturbée avec une perturbation appartenant à l'ensemble des perturbations admissibles.*[67]

$$\hat{R}_{adv}(h_{\theta}, D) = \frac{1}{|D|} \sum_{(x,y) \in D} \max_{\delta \in \Delta(x)} \ell(h_{\theta}(x + \delta), y)$$

3.10 Apprentissage Adversaire

L'apprentissage adversaire est un ensemble de méthodes de défense contre les attaques adversaires. Ces méthodes consistent à introduire des exemples adversaires durant l'apprentissage afin que le modèle soit robuste aux attaques.

L'objectif de cet apprentissage est de déterminer les paramètres θ qui minimisent le risque adversaire R [67].

$$\underset{\theta}{\text{minimize}} \hat{R}_{adv}$$

L'optimisation de θ s'effectue à l'aide de la descente du gradient (dans ce cas une descente de gradient sur un minibatch B) :

$$\theta = \theta - \frac{\alpha}{|B|} \sum_{x,y \in B} \nabla_{\theta} \max_{\delta \in \Delta(x)} \ell(h_{\theta}(x + \delta), y)$$

Le gradient peut s'écrire :

$$\nabla_{\theta} \max_{\delta \in \Delta(x)} \ell(h_{\theta}(x + \delta), y) = \nabla_{\theta} \ell(h_{\theta}(x + \delta^*(x)), y)$$

avec :

$$\delta^*(x) = \operatorname{argmax}_{\delta \in \Delta(x)} \ell(h_{\theta}(x + \delta), y)$$

L'algorithme d'apprentissage adversaire à l'aide d'exemples adverses est alors introduit :

Algorithm 4 Algorithme d'apprentissage adversaire par entraînement adversaire[67]

- 1: g : vecteur de gradient
 - 2: α : Taux d'apprentissage
 - 3: **Répéter** :
 - 4: Selectionner un mini-batch B et initialiser le vecteur gradient à 0
 - 5: **Pour chaque** $(x, y) \in B$
 - 6: Calculer une approximation de δ^* en optimisant :
 - 7: $\delta^*(x) = \operatorname{argmax}_{\delta \in \Delta(x)} \ell(h_{\theta}(x + \delta), y)$
 - 8: Ajouter le gradient de $x + \delta^*(x)$
 - 9: $g := g + \nabla_{\theta} \ell(h_{\theta}(x + \delta^*(x)), y)$
 - 10: Mettre à jour les paramètres θ
 - 11: $\theta = \theta - \frac{\alpha}{|B|} g$
 - 12: **Jusqu'à condition d'arrêt**
-

3.10.1 Entraînement adversaire pour la classification de texte

Miyato [68] propose une méthode d'entraînement adversaire pour une tâche de classification de texte sur un réseau LSTM. La perturbation r_{adv} pour un input x est définie :

$$r_{adv} = \operatorname{argmax}_{r, ||r|| \leq \epsilon} \log p(y|x + r, \hat{\theta})$$

avec :

$\hat{\theta}$ est un ensemble de paramètres constants, c'est à dire fixé lors de la procédure de génération de l'exemple adversaire. Une approximation de r_{adv} est introduite :

$$r_{adv} = -\epsilon \frac{g}{||g||_2}$$

$$g = \nabla_x \log p(y|x, \hat{\theta})$$

avec g le vecteur gradient de l'entrée x , $||g||_2$ la norme du vecteur gradient et ϵ un paramètre empirique.

Soit s le vecteur résultant de la concaténation des plongements lexicaux d'une séquence de mots $s = [v^{(1)}, v^{(2)}, \dots, v^{(T)}]$, y_n le label de l'entrée s et N le nombre d'exemples.

L'erreur adversaire est définie :

$$L_{adv}(\theta) = -\frac{1}{N} \sum_{n=1}^N \log p(y_n | s_n + r_{adv}, \theta)$$

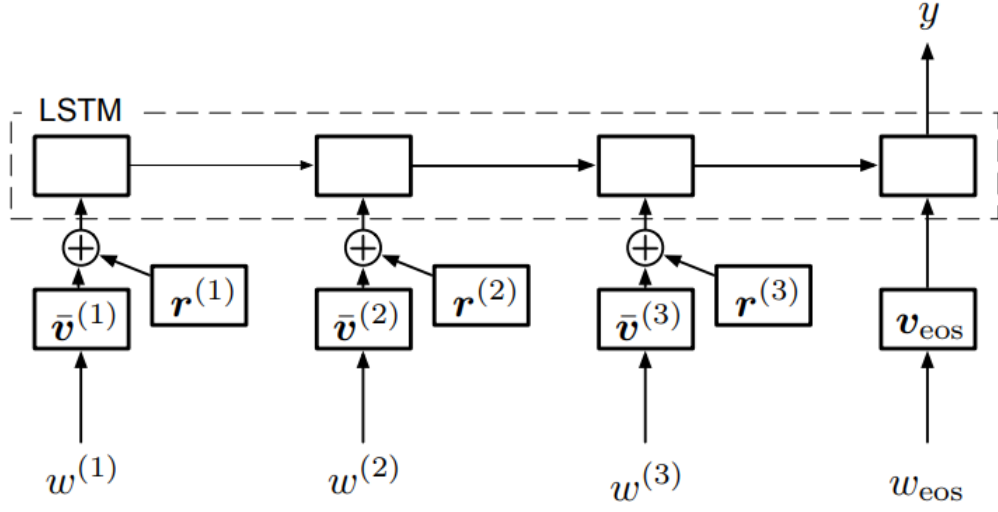


FIGURE 3.18 – Architecture du réseau adversaire

3.10.2 Entraînement adversaire virtuel pour la classification de texte

Similairement, l'entraînement adversaire virtuel pour la classification de texte est proposé par Miyato[68]. A la différence de l'entraînement adversaire, cette méthode ne requiert pas la présence de labels, étendant ainsi la méthode d'entraînement adversaire à des problèmes semi-supervisés. La fonction d'entropie relative est utilisée lors du calcul de l'erreur :

$$\text{KL}[p(\cdot|x, \theta) || p(\cdot|x + r_{v-adv}, \theta)]$$

Lors de l'apprentissage, une approximation de $p(\cdot|x, \theta)$ est calculée pour chaque exemple non étiqueté. Cette approximation n'est pas naïve, car elle devient proche de $p(\cdot|x, \theta)$ lorsque le nombre des exemples étiquetés est grand[69]. Une approximation de la perturbation adversaire virtuelle est aussi calculée :

$$r_{v-adv} = -\epsilon \frac{g}{||g||_2}$$

$$g = \nabla_{s+d} \text{KL}[p(\cdot|s, \hat{\theta}) || p(\cdot|s + d, \hat{\theta})]$$

avec d un vecteur de T -dimensionnel aléatoire.

L'erreur adverse virtuelle est alors égale à la somme des entropies relatives entre la probabilité d'obtenir la sortie de chaque entrée $s_{n'}$ et la probabilité d'obtenir la sortie (appelée sortie virtuelle pour les exemples non étiquetés) de son entrée perturbée $s_{n'} + r_{v-adv;n'}$ pour tous les exemples (étiquetés et non étiquetés) de l'ensemble d'apprentissage. La formule de l'erreur adverse virtuelle est définie par :

$$L_{v-adv}(\theta) = \frac{1}{N'} \sum_{n=1}^{N'} \text{KL}[p(\cdot|s_{n'}, \theta) || p(\cdot|s_{n'} + r_{v-adv;n'}, \theta)]$$

avec N' le nombre d'exemples étiquetés et non étiquetés.

3.11 Mesures de performance

La méthodologie d'évaluation des systèmes de prédiction depuis Salton[70], s'appuie sur les mesures suivantes :

3.11.1 Précision

La précision est une mesure de performance exprimant la proportion de réponses positives correctement prédites par le modèle parmi l'ensemble des réponses positives retournées du modèle.

La précision se calcule comme suit :

$$P = \frac{TP}{TP + FP}$$

avec :

TP (Vrai positif) : Nombre d'exemples où le modèle prédit correctement la classe positive.

FP (Faux positif) : Nombre d'exemples où le modèle prédit incorrectement la classe positive.

La précision ne permet pas d'exprimer de manière fiable la performance d'un système lorsque les classes des données sont déséquilibrées (le nombre d'exemples appartenant à une classe est beaucoup plus grand que le nombre d'exemples appartenant à l'autre classe.)

3.11.2 Rappel

Le rappel mesure la proportion de réponses positives d'un modèle parmi l'ensemble des réponses réellement positives. Le rappel se calcule comme suit :

$$R = \frac{TP}{TP + FN}$$

avec :

FN (Faux négatif) : Nombre d'exemples où le modèle prédit correctement la classe négative.

3.11.3 F-score

Le F-score, ou f-mesure, est une moyenne harmonique⁷ de la précision et du rappel. Sa formule est définie par :

$$\text{F-mesure} = \frac{2P * R}{P + R}$$

3.11.4 Exactitude (Accuracy)

L'exactitude ou justesse est la quantité exprimant la proportion de réponses correctement prédites par le modèle parmi l'ensemble des réponses retournées du modèle. L'exactitude se calcule comme suit :

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

3.11.5 Coefficient de Corrélacion de Matthews (MCC)

Cette mesure est utilisée pour les classifications et a l'avantage d'être une mesure fiable même lorsque les classes de l'ensemble de données sont déséquilibrées. Le coefficient de corrélation de Matthews se calcule à l'aide de la formule suivante :

$$MMC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

avec :

TN (Vrai négatif) : Nombre d'exemples où le modèle prédit correctement la classe négative.

7. la moyenne harmonique est l'inverse de la moyenne arithmétique des inverses des termes.

Conception

4.1 Introduction :

Motivés par le succès des réseaux de neurones dans un large éventail d'applications liées au profilage, nous proposons une application des réseaux de neurones pour modéliser la prédiction des traits de personnalité des individus en analysant leurs publications textuelles sur les réseaux sociaux.

4.2 Conception Générale du Système de Prédiction de Personnalité

Nous proposons pour le problème de prédiction des traits de personnalité sur les réseaux sociaux, une approche basée sur les plongements lexicaux et les réseaux de neurones. Notre approche tente d'exploiter l'hypothèse lexicale à travers la représentation du texte sous forme de plongement lexical.

Nous commençons par prétraiter notre corpus brut. S'en suivra, la transformation des tokens en vecteurs numériques (word embeddings) selon une architecture spécifique (Glove, Word2Vec, FastText). Les vecteurs ainsi obtenus seront donnés en entrée à une architecture de réseaux de neurones qui produit en sortie le degré d'un des 5 traits de personnalité, comme sur la figure 4.1. La structure générale du modèle est la même pour la prédiction de tous les traits. La seule différence réside dans le trait à prédire.

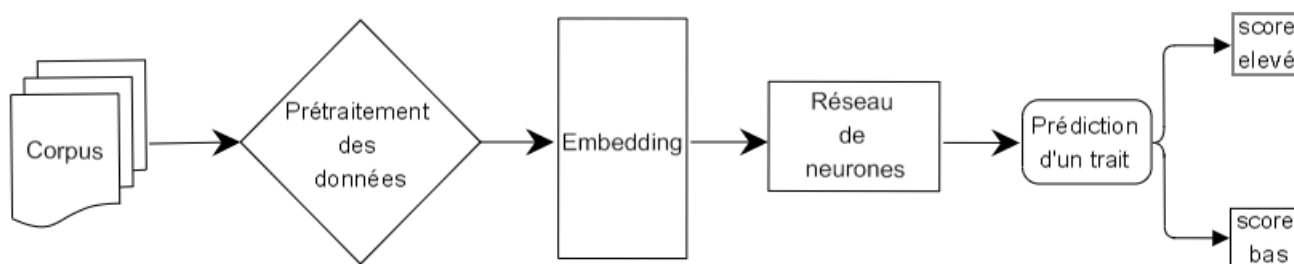


FIGURE 4.1 – Schéma général du système de prédiction de personnalité

4.3 Préparation du Corpus

La première étape de notre étude, qui consiste à choisir le corpus d'apprentissage, le nettoyer et la préparation des données, mérite une attention particulière.

4.3.1 Description du Corpus d'Apprentissage :

L'échantillon *myPersonality*, comme présenté dans le chapitre précédent, est un corpus qui regroupe les statuts de 250 utilisateurs Facebook, comprenant près de 10000 status. A chaque utilisateur est associé un score (élevé ou bas) d'un trait de personnalité selon le modèle du Big Five. Le corpus, en dépit de son volume réduit, reste très représentatif de la population. En effet, nous avons étudié sa diversité pour mieux le comparer aux études traditionnelles faites par les psychologues de la personnalité.

1. **Genre :** Selon Vazire et al.[71], 71% des participants aux études sur la personnalité, et plus particulièrement le modèle des Big Five sont de sexe féminin. Notre échantillon, est représenté par 63% de femmes, comme nous pouvons le voir dans la figure 4.2.
2. **Age :** La population de Facebook est encore majoritairement jeune, tout comme les participants de *myPersonality*. L'âge moyen des utilisateurs est de 23,5 ans et près de la moitié d'entre eux (47%) ont entre 18 et 24 ans (Figure 4.2). Néanmoins, l'âge des participants de l'échantillon *myPersonality* semble être très similaire à celui représenté par les études traditionnelles faites par les psychologues. Comme estimé par Vazire et al.[71], l'âge moyen dans les échantillons est de 23 ans.
3. **Couverture géographique :** Les réseaux sociaux et le web en général ne connaissent pas les frontières, par conséquent, les études en ligne sont accessibles à des populations géographiquement éloignées. *myPersonality*, bien qu'ayant vu le jour aux États-Unis, compte plus de 42% de répondants à l'extérieur des États-Unis. (à noter que l'application *myPersonality* était exclusivement disponible en langue anglaise). Plus spécifiquement, 44 pays sont représentés par plus de 1 000 utilisateurs Facebook.

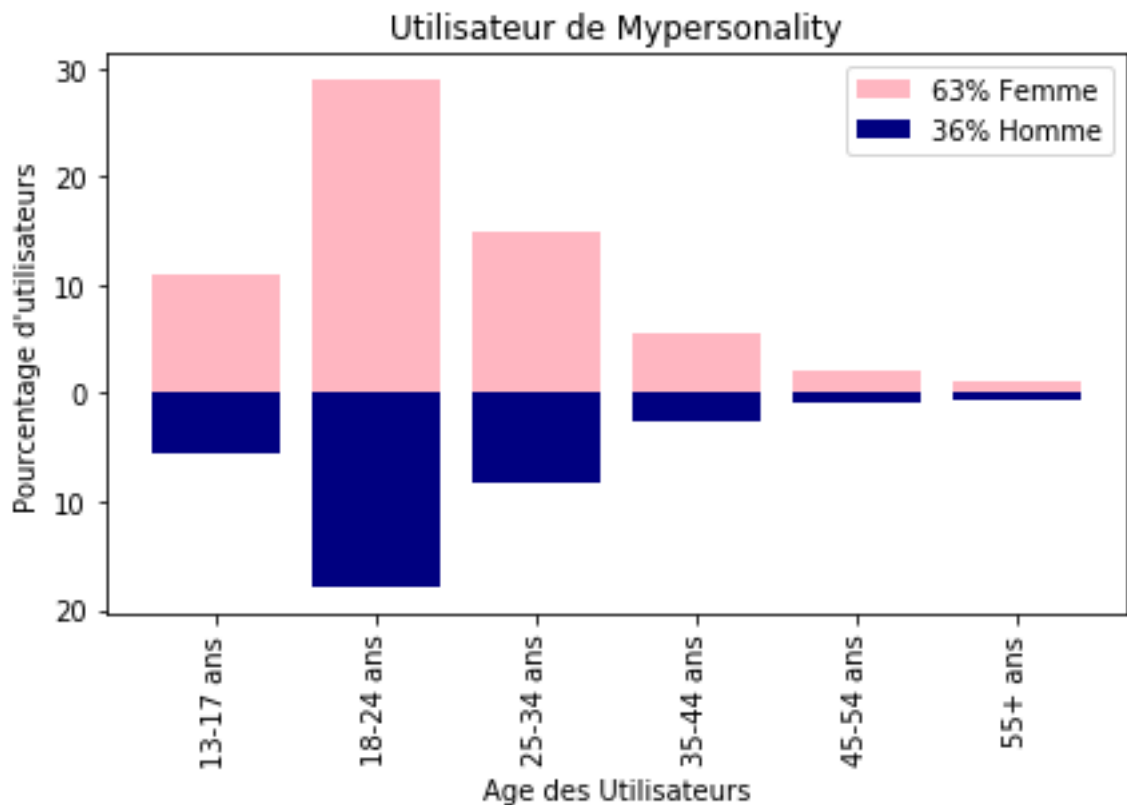


FIGURE 4.2 – Distribution des utilisateurs (selon le genre et l'âge) dans le corpus myPersonality

Toutes ces caractéristiques font de l'échantillon *myPersonality* un corpus fiable pour une étude systématique sur les traits de la personnalité.

La distribution de l'ensemble de données (utilisateurs) *myPersonality* en fonction du trait de personnalité est présentée dans le tableau ci-dessous.

Distribution du Dataset *myPersonality* (selon les traits des utilisateurs) :

Trait \ Score	OPN	CON	EXT	AGR	NEU
Elevé	176	130	96	134	99
Bas	74	120	154	116	151

Le principal inconvénient de ce corpus est son volume, il est composé de 9917 statuts Facebook, ce qui correspond à 92788 mots dont 37% sont des mots vides (qui n'apportent rien au sens) et 1024 de mots étrangers (inexistant dans la langue anglaise, comprenant langage familier et entités nommées). En outre, 4.5% des statuts sont trop petits (messages trop courts) pour en extraire de l'information pertinente.

Un second inconvénient est son caractère non homogène. On constate de la figure 4.3, que le trait de personnalité OPN (« Ouverture ») a 7370 statuts représentant la classe ‘score élevé’ (1) contre 2547 seulement dans la classe ‘score bas’ (0). De même pour le trait NEU.

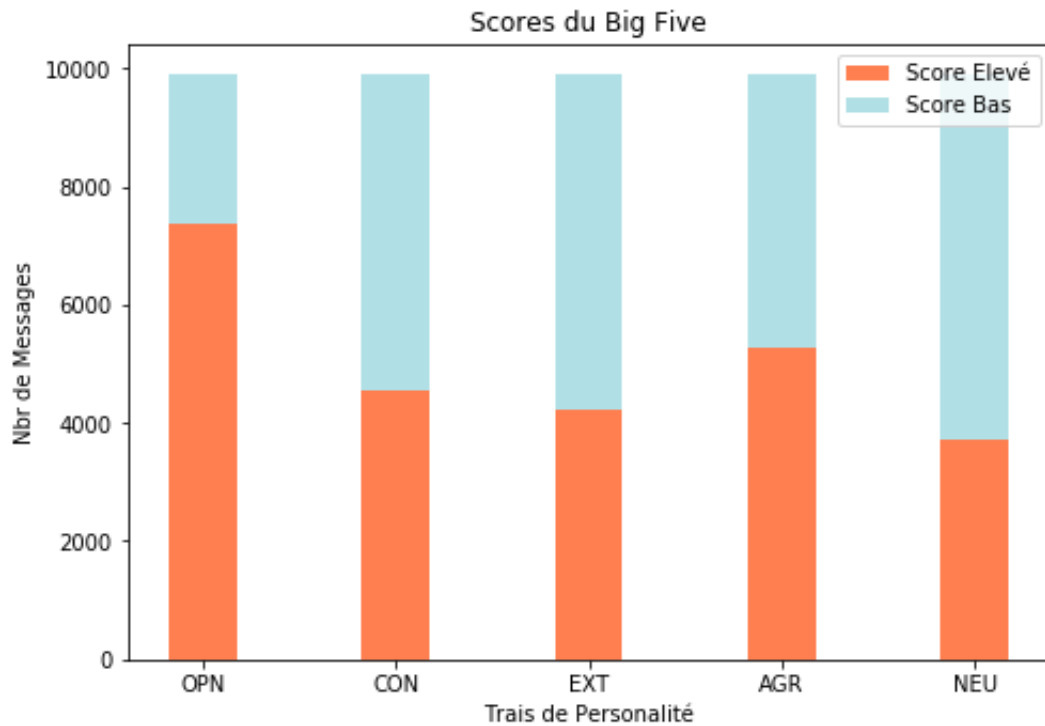


FIGURE 4.3 – Distribution des classes (scores) pour chaque statut

Trait \ Classe	OPN	CON	EXT	AGR	NEU
Score élevé	7370	4556	4210	5268	3717
Score bas	2547	5361	5707	4649	6200

La plupart des systèmes de classification suppose que les classes sont équilibrées et, par conséquent, construisent la fonction d’erreur correspondante pour maximiser un taux de précision global. Dans le cas d’un corpus non homogène, le résultat conduirait à des prédictions faussées.

Ainsi, il n’est pas envisageable d’effectuer l’apprentissage sans considérer un rééquilibrage de la distribution des données. Les approches les plus communes utilisent le principe d’échantillonnage : soit l’on considère un sous-échantillonnage des objets de la classe majoritaire, soit l’on considère un sur-échantillonnage de la classe minoritaire. Les deux approches essaient de rééquilibrer la distribution des classes en termes d’instances incluses dans l’apprentissage.

Afin de répondre aux défis posés par l'ensemble d'apprentissage, nous nous proposons d'implémenter un système d'apprentissage semi-supervisé. Pour cela il nous a fallu un grand ensemble de données textuelles non étiquetées. Etant donné les scandales entourant Facebook, notamment celui de Cambridge Analytica, il est devenu extrêmement difficile d'avoir accès à des corpus de statuts facebook ou encore d'en faire l'extraction à l'aide d'outils.¹

Nous nous sommes donc tournés vers un autre média. Reddit a récemment dépassé Facebook pour devenir le 3ème site le plus visité aux États-Unis. Le dataset que nous utilisons a été collecté dans le cadre de la classification de texte [72], et comporte un peu plus de 1 million de publications, ce qui représente 171 millions de tokens.

4.3.2 Prétraitement et Représentation du texte :

L'information diffusée dans les médias sociaux est hautement complexe, dû à son caractère informel.

En conséquence, l'application des méthodes habituelles de TAL dans ce contexte ne se fait pas sans difficulté.

Après avoir esquissé les approches générales du prétraitement de nos données textuelles, nous allons décrire maintenant les méthodes particulières qui permettent de répondre à la problématique qui se pose.

1. Normalisation des liens :

Dans notre contexte, un lien dans une publication n'a pas un poids sémantique, nous avons donc décidé de le remplacer par des balises afin de ne pas perdre l'information tout en atténuant son poids.

Exemple :

"Follow me on <http://www.youtube.com/watch?v=yPJRPs> " → "Follow me
on <url> "

1. Le scandale Facebook-Cambridge Analytica ou la fuite de données Facebook-Cambridge Analytica renvoie aux données personnelles de 87 millions d'utilisateurs Facebook que la société Cambridge Analytica a commencé à recueillir dès 2014.

2. Anonymisation des identifiants utilisateurs :

Nous avons implémenté ce type de nettoyage, pour tout d'abord garantir l'anonymat des utilisateurs et aussi car un identifiant n'est pas un mot clé dans notre contexte. Le nom d'utilisateur n'amène aucune information pertinente quant à la prédiction de ses traits de personnalité. D'autres parts, il peut engendrer des similarités entre les messages d'un même utilisateur et par conséquent biaiser l'apprentissage.

Exemple :

"I guess you're right `Nathaniel_b` " \longrightarrow "I guess you're right `<user>` "

Suivant les mêmes raisons, nous avons appliqué le même nettoyage pour les 'email', 'adresse' et 'numéro de téléphone'.

3. Traitement des pictogrammes (emojis) :

Cette tâche est la plus ardue, car un pictogramme est une représentation typographique et ses déclinaisons sont diverses et variées, affranchies de toutes règles syntaxiques. Un autre problème posé par les pictogrammes est qu'ils peuvent être composés uniquement de lettres telles que 'xp' (qui représente un visage rieur) ou encore sa forme dérivée 'xpppp' qui doit être distinguée de la sous-chaîne d'un mot tel que 'explain'.

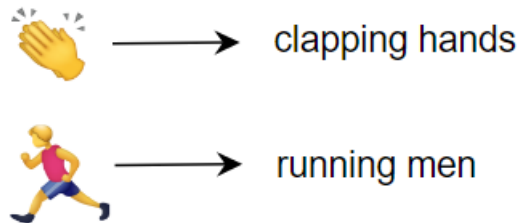
Ou encore '<3' qui dans ce contexte est un cœur, mais dans celui-ci ' $1 + 2x < 3$ ' représente une inéquation.

Une autre particularité des pictogrammes est qu'ils dépeignent une palette de sentiments très larges et souvent opposés, il nous est donc paru important de différencier quatre types d'émotions, par le biais de balises qui sont `<sadface>`, `<smiley>`, `<lolface>` et `<neutralface>`. (Exemple 1)

Aussi, pour les pictogrammes les plus complexes, qui ne peuvent pas être réduits à des types d'émotions, nous décidons de les remplacer par leur explication (Exemple 2).

Exemple 1 :

” It was so funny xDDDDDD ” → ” It was so funny <smiley> ”

Exemple 2 :**4. Suppression de la ponctuation :**

Cela consiste à nettoyer les mots par la suppression des guillemets, des virgules, des points, des points d’exclamation et des points d’interrogation, etc.

Exemple :

” What do you think about “HIM”????! ” → ” What do you think about
HIM ”

Cette étape a permis de réduire le volume du corpus d’un dixième.

5. Suppression des caractères numériques :

Les nombres étant inutiles pour le profilage, nous les avons remplacés par la balise <number>

Exemple :

” I got 20 on my test ” → ” I got <number> on my test ”

6. Transformation des majuscules en minuscules :

Pour résoudre le problème des majuscules et des minuscules nous avons décidé d'appliquer deux types de traitements, le premier est la conversion de tous les mots en minuscule. Le deuxième traitement, est l'ajout d'une balise `<allcaps>` dans le cas où toutes les lettres du mots sont en majuscule afin de garder une information sur le style d'écriture de la personne (généralement les mots tout en majuscules expriment une vocifération ou permettent de mettre l'accent sur un mot).

Exemple :

" FOOTBALL!!!! It's gonna be awesome!!!" → "football!!!!<allcaps> It's gonna be awesome!!!"

7. Traitement des répétitions de lettres :

Le langage utilisé sur les réseaux sociaux est pour la plupart du temps, familier. Il n'est donc pas rare que des mots soient écrits avec une lettre (ou plusieurs) qui se répète alors qu'elle ne le devrait pas. Par exemple le mot "dog" peut se retrouver sous la forme "dooooooooog". Lorsqu'un mot contient des lettres identiques qui se répètent plus de deux fois, elles sont remplacées par une occurrence de cette même lettre puis est ajoutée la balise `<elong>` afin de ne pas perdre cette caractéristique du type de l'écriture de l'individu.

Exemple :

" I saw a cute doooooooooog yesterday " → " I saw a cute dog <elong> yesterday"

8. Traitement des répétitions de mots :

Tout comme pour les lettres répétées, les mots qui se répètent sont très fréquents dans le langage non formel, nous trouvons par exemple : yesyesyes. Comme pour les lettres répétées, nous ne gardons qu'une occurrence, puis nous ajoutons la balise `<repeat>` pour préserver l'information capturée.

Exemple :

"See you. byebye " → " See you. bye <repeat> "

Toutes les étapes de prétraitement détaillées ci-dessus sont effectuées à l'aide d'expressions régulières.

9. Correction orthographique :

L'étape de la correction d'orthographe s'est imposée après avoir constaté que près de 15% des mots composant le corpus étaient mal orthographiés, et finiraient inmanquablement remplacés par la balise <unknown> dans le cas d'utilisation d'un modèle déjà entraîné pour l'étape du plongement lexical.

Nous avons utilisé pour cela des algorithmes de correction automatique implémentant la distance de Levenshtein. La distance de Levenshtein mesure la similarité entre deux chaînes de caractères. Elle est égale au nombre minimal de caractères qu'il faut supprimer, insérer ou remplacer pour passer d'une chaîne à l'autre. (distance d'Édition). C'est donc un peu moins de 15000 mots qui ont été corrigés.

Exemple :

"If your friend got **assassined** , how would you respond" → "if your friend got **assassinated**, how would you respond"

10. Tokenisation :

Enfin, nous avons représenté chaque message en liste d'unités linguistiques.

Exemple :

"today sunny day" → " **[today]** **[sunny]** **[day]**"

Afin de ne pas perturber l'apprentissage avec des messages trop courts pour contenir de l'information pertinente, nous avons supprimé toutes les publications contenant moins de 3 unités linguistiques (tokens).

11. Traitement des mots vides :

Au cours de ce prétraitement, nous avons fait le choix de conserver les mots vides, ces entités représentent 0.04% de la taille du corpus total. L'importance des mots vides notamment des pronoms a été mise en évidence dans différentes recherches comme ceux de Dewaele [73] et Holtgraves [74] qui montrent la relation entre l'usage fréquent des pronoms de la première personne du pluriel ('we') chez les personnes avec un score élevé pour le trait EXT. Similairement, un usage fréquent du pronom de la 2ème personne du singulier ('you') serait corrélé à un score élevé pour le trait CON selon Mehl [75].

4.3.3 Plongement Lexical

Comme nous l'avons vu dans le chapitre précédent, la méthode GloVe [76] (Pennington et al., 2014) développée par Stanford repose sur l'utilisation de cooccurrences entre les termes (le nombre de fois qu'un terme apparaît en concomitance avec un autre).

A la différence de Word2Vec, qui est une méthode prédictive, GloVe prend en compte l'information portée par l'entière du corpus et non sur une fenêtre de mots uniquement, d'où le nom GloVe, pour Vecteur Global.

Il devient possible de reproduire exactement les mêmes performances que Word2Vec et de FastText et ce de manière bien plus rapide. En effet, le caractère statistique de GloVe permet une implémentation parallèle, offrant un apprentissage plus rapide. [77]

Dans cette partie, nous nous sommes intéressés à l'étude de l'impact du plongement de mots GloVe sur les performances de notre système. Les modèles de plongements de mots utilisés dans cette étude sont les suivants :

- **Glove Personnalisé :**

Nous avons entraîné GloVe (Pennington et al., 2014) [76] sur notre corpus d'apprentissage. La taille du vocabulaire du corpus *myPersonality* est de 50.000 mots, chaque plongement lexical de mot a un nombre de caractéristiques (features) égal à 300 dimensions.

Pour le choix du nombre de dimensions nous nous sommes référés à l'étude de Landauer et Dumais [78] qui ont montré, avec des expériences empiriques, que 300 est le nombre optimal de caractéristiques pour des représentations de mots.

Toutefois, n'ayant accès qu'à un corpus de taille réduite, avec un vocabulaire de 50.000 mots seulement contre un vocabulaire de 1.2 millions de mots pour les modèles généralement utilisés lors de la représentation de textes issus de réseaux sociaux. Notre modèle ne pourra vraisemblablement pas représenter un pourcentage potentiellement important de termes présents dans nos données test.

- **Glove Pré-entraîné :**

Nous avons donc utilisé des vecteurs pré-entraînés avec GloVe, disponible à [76].

Les vecteurs que nous avons utilisé ont été entraînés sur 2 milliards de tweets avec une taille de vocabulaire égale à 1.2 millions et un nombre de caractéristiques égal à 200 dimensions.

4.4 Classifieur pour la prédiction des traits de personnalité

Nous procédons à l'implémentation et à la conception de différentes méthodes d'apprentissage automatique, communément utilisées pour des tâches de classification de texte, sur le dataset choisi (*myPersonality*).

4.4.1 Réseaux récurrents

Pour les réseaux récurrents tels que les réseaux récurrents à mémoire court et long terme (*LSTM*), à portes (*GRU*) et bi-directionnels (*Bi-RNN*), nous proposons l'architecture suivante :

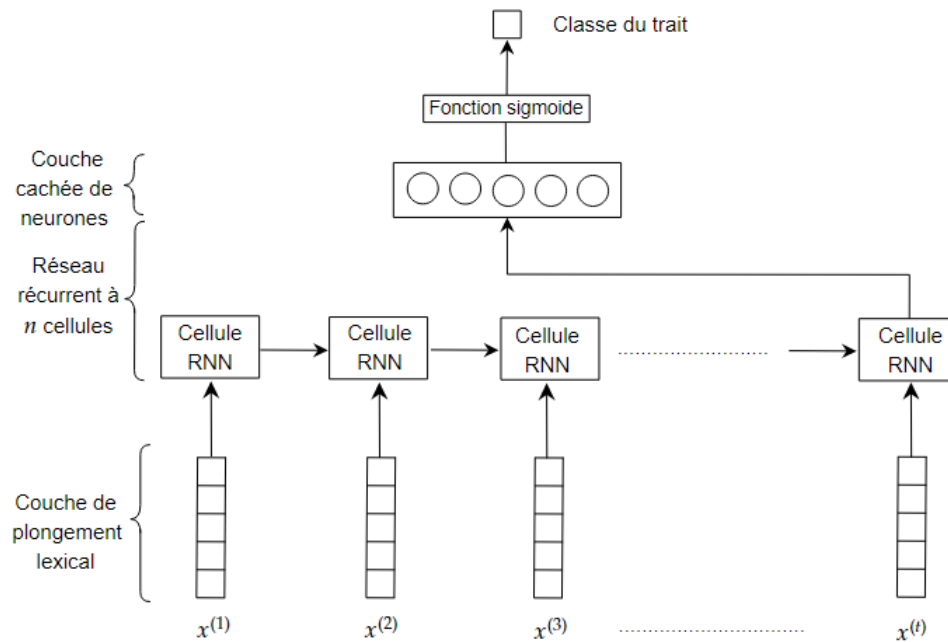


FIGURE 4.4 – Schéma de l'architecture pour les réseaux récurrents

Les données sont passées à une couche d'embedding (GloVe) d'abord puis à un réseau récurrent de n cellules récurrentes, les données résultantes sont ensuite passées à une couche cachée. Enfin la couche de sortie sera composée d'un neurone unique avec avec une fonction d'activation sigmoïde.

4.4.1.a LSTM

La capacité des réseaux LSTM à capturer les dépendances longues distances dans les données séquentielles est un atout majeur pour la résolution des problèmes de classification de textes. En effet, on retrouve de nombreuses dépendances dans les données textuelles. Les avantages de ces réseaux et leur efficacité dans la classification de textes nous poussent à penser qu'ils seraient intéressants pour notre problème.

4.4.1.b GRU

Les réseaux GRU sont une version simplifiée des réseaux LSTM. Ils permettent de conserver les propriétés des réseaux LSTM. Les différences de performances entre les réseaux GRU et LSTM dépendent du problème, il n'est à priori pas possible de privilégier un modèle à l'autre.

4.4.1.c RNN Bi-directionnel

Ce type de réseaux permet de capturer plus d'informations qu'un réseau récurrent classique unidirectionnel tel que LSTM ou GRU grâce à sa capacité à traiter les données futures d'une étape t .

4.4.1.d Réseaux récurrents adversaires

Nous avons fait le choix de tester les réseaux LSTMs adversaires car la génération d'exemples adversaires lors de la phase d'entraînement permet de développer un modèle plus robuste qu'un réseau LSTM classique. Cette propriété de robustesse au bruit est intéressante en raison du nombre limité de données dont dispose le corpus *myPersonality*. Similairement, les réseaux LSTM adversaires virtuels ont également prouvé leur robustesse pour des tâches de classification de texte mais proposent aussi d'étendre la classification à des données non étiquetées. En raison de la difficulté d'obtenir les résultats des tests NEO sur des internautes et de construire un corpus du même type que *myPersonality*, une approche semi-supervisée est pertinente pour contourner cette limitation puisque les données textuelles non étiquetées sont plus facilement extractibles. Plus souvent appliquées en informatique visuelle, les méthodes adversaires et adversaires virtuelles sont relativement récentes en classification de texte, et plus précisément en analyse de sentiments (Ian Goodfellow 2017)[14]. À notre connaissance, l'application de ces techniques pour le problème de prédiction des traits de personnalité sur les réseaux sociaux est inédite.

4.4.2 Choix des hyperparamètres

Les réseaux de neurones profonds ont de multiples hyper-paramètres à optimiser notamment l'architecture (nombre de couches, taille de ces couches, type, agencement...) et les paramètres de l'apprentissage liés à cette architecture. Les performances du modèle d'apprentissage dépendent du bon choix de ces paramètres. Dans le cadre d'un projet complexe, s'appuyer sur des intuitions ne suffit plus. Nous avons donc utilisé un système de recherches des hyper paramètres les plus optimaux, exploitant des algorithmes de recherche hyperband, une variation de la recherche aléatoire. [79]

Une propriété des réseaux de neurones est que leur apprentissage est généralement itératif, une variante de descente du gradient. Par conséquent, il est possible d'interrompre l'apprentissage à tout moment, d'évaluer le réseau, puis de reprendre la phase d'apprentissage. L'Hyperband [79] tire parti de cette caractéristique propre aux réseaux de neurones. Son principe est simple : sélectionner aléatoirement un groupe de paramètres à partir d'une distribution uniforme, faire un apprentissage partiel, évaluer le modèle, puis reprendre l'apprentissage avec les paramètres les plus optimaux, et répétez le cycle jusqu'à épuisement des ressources disponibles.

4.4.3 Hyperparamètres des réseaux récurrents :

Après la recherche par expérimentation des meilleurs paramètres, la fonction d'apprentissage choisie est la fonction Adam, la taille des mini-batch est fixée à 64. Voici la liste des hyperparamètres pour les architectures pour chacun des 5 traits :

- Trait d'ouverture à l'expérience (OPN) :

<div>Réseau</div> <div>Paramètre</div>	LSTM	GRU	Bi-RNN	LSTM-ADV	LSTM-V-ADV
Nombre de cellules récurrentes	1024	512	128	1024	1024
Nbre de neurones couche cachée	30	70	10	30	30
Taux d'apprentissage	0.0001	0.01	0.001	0.0001	0.0001

- Trait Conscientieux (CON) :

<div> Réseau </div> <div>Paramètre</div>	LSTM	GRU	Bi-RNN	LSTM-ADV	LSTM-V-ADV
Nombre de cellules récurrentes	256	256	64	256	256
Nbre de neurones couche cachée	70	30	64	70	70
Taux d'apprentissage	0.001	0.01	0.01	0.001	0.001

- Trait Extraversion (EXT) :

<div> Réseau </div> <div>Paramètre</div>	LSTM	GRU	Bi-RNN	LSTM-ADV	LSTM-V-ADV
Nombre de cellules récurrentes	512	256	64	512	512
Nbre de neurones couche cachée	50	90	64	50	50
Taux d'apprentissage	0.0001	0.001	0.001	0.0001	0.0001

- Trait Amabilité (AGR) :

<div> Réseau </div> <div>Paramètre</div>	LSTM	GRU	Bi-RNN	LSTM-ADV	LSTM-V-ADV
Nombre de cellules récurrentes	768	768	64	768	768
Nbre de neurones couche cachée	80	60	64	80	80
Taux d'apprentissage	0.001	0.001	0.001	0.001	0.001

- Trait Névrosisme (NEU) :

<div> Réseau </div> <div>Paramètre</div>	LSTM	GRU	Bi-RNN	LSTM-ADV	LSTM-V-ADV
Nombre de cellules récurrentes	768	768	64	768	768
Nbre de neurones couche cachée	60	60	64	60	60
Taux d'apprentissage	0.001	0.001	0.001	0.001	0.001

4.4.4 Réseaux Convolutifs

Régulièrement utilisés pour la classification d'images, les réseaux convolutifs peuvent également être adaptés à la classification de textes. Les réseaux convolutifs sont efficaces pour la détection de caractéristiques dans les textes mais peinent à capturer les dépendances contextuels à l'instar des réseaux LSTM.

Les données en entrée prennent la forme d'une matrice $n \times d$ avec n la taille de chaque phrase et d la taille du plongement lexical. Chaque ligne correspond au plongement lexical de chaque mot de la phrase. La matrice est passée à une ou plusieurs couches de convolution. Chaque couche de convolution est suivie d'une couche de pooling max. Les données résultantes sont alors passées à une couche cachée fully connected avec une fonction d'activation ReLu. La couche de sortie consiste en un neurone unique avec une fonction d'activation sigmoïde. La figure suivante représente l'architecture générale du réseau CNN :

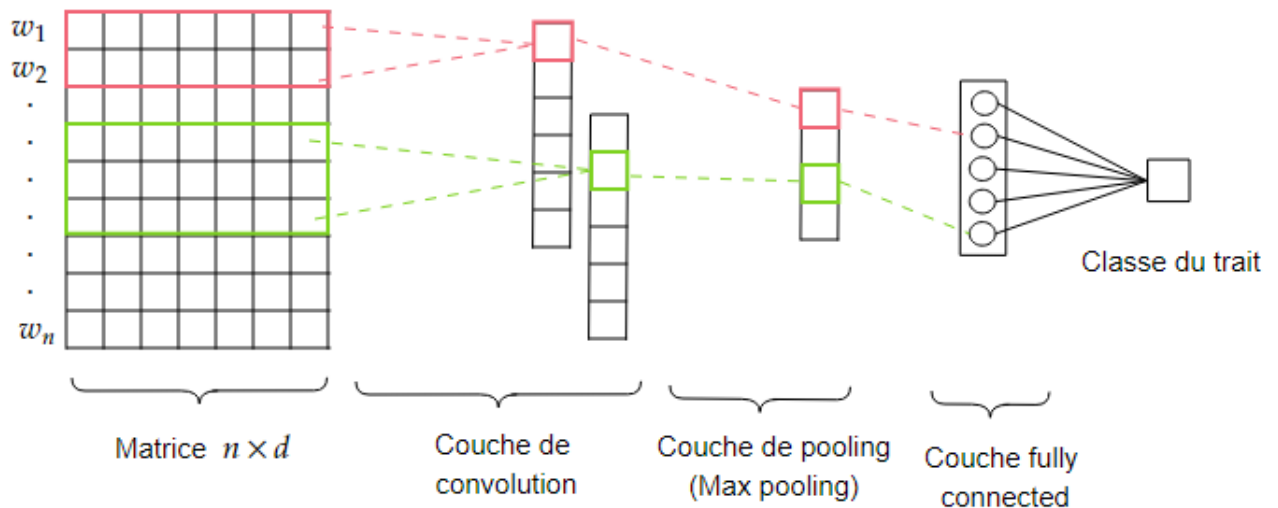


FIGURE 4.5 – Schéma de l'architecture du réseau CNN

4.4.5 Hyperparamètres du Réseau Convolutif :

Après la recherche par expérimentation des meilleurs paramètres. nous obtenons la liste des hyperparamètres pour les architectures pour chacun des 5 traits :

Trait Paramètre	Ouverture	Conscientieux	Extraversion	Amabilité	Névrosisme
Nbre de couches de convolution	4	4	4	4	4
Nbre de neurones fully connected	128	64	128	128	128
Taille des filtres de convolution	2-3-4-5-6	2-3-4-5-6	2-3-4-5-6	2-3-4-5-6	2-3-4-5-6

Tests et Expérimentations

5.1 Introduction

Dans ce chapitre, nous procédons aux tests des modèles proposés dans le chapitre précédent. Nous discutons les résultats obtenus et nous les comparons avec d'autres approches proposées.

5.1.1 Méthodologie

Le dataset *myPersonality* étant prétraité suivant les étapes décrites dans le chapitre conception, nous le divisons en 70% pour la phase d'entraînement (train set), 30% (test set). 15% de l'ensemble d'entraînement sera réservé à la validation. La taille des vecteurs d'entrée (phrases) choisie est égale à la valeur médiane des tailles de statuts qui est de 30 mots. Ensuite, nous procédons à l'apprentissage des modèles avec leurs architectures respectives. Nous implémentons lors de ces apprentissages, un arrêt prématuré en cas de stagnation du modèle.

5.1.2 Environnement d'expérimentation

Le prétraitement du corpus sera effectué sous **Python**[80], à l'aide de la bibliothèque **nltk**[81]. L'implémentation des modèles présentés sera programmée sous **Tensorflow**[82] et **Keras**[83]. Les modèles proposés nécessitent des puissances de calculs conséquentes, nous utiliserons donc les plateformes et spécifications techniques suivantes :

Plateforme	Google Colab	Microsoft Azure
GPU	Tesla K80 (2496 CUDA cores)	Tesla K80
RAM	12.6 GB	56 GB
Taille disque	100 GB	300 GB

5.2 Tests comparatifs

Nous mesurons la performance de nos modèles sur la base de la précision et le f-score durant la phase de tests

Voici les résultats obtenus :

Modèle \ Trait		Ouverture	Conscientieux	Extraversion	Amabilité	Névrosisme
LSTM	Acc	0.64	0.58	0.67	0.57	0.60
	F-Score	0.70	0.43	0.42	0.6	0.46
LSTM-ADV	Acc	0.65	0.67	0.72	0.60	0.65
	F-Score	0.73	0.71	0.56	0.63	0.62
CNN	Acc	0.66	0.75	0.50	0.64	0.60
	F-Score	0.70	0.68	0.60	0.68	0.70
GRU	Acc	0.53	0.57	0.50	0.57	0.59
	F-Score	0.70	0.42	0.20	0.60	0.50
Bi-RNN	Acc	0.64	0.70	0.68	0.67	0.65
	F-Score	0.69	0.67	0.53	0.67	0.64
LSTM-V-ADV	Acc	0.58	0.62	0.61	0.58	0.53
	F-Score	0.45	0.58	0.55	0.57	0.59

5.2.1 Discussion des résultats

Ce tableau rend compte des résultats des performances de chacun des modèles pour chacun des 5 traits (OPN, CON, EXT, AGR, NEU) durant la phase de test. En comparant les performances du réseau *LSTM* et *LSTM-ADV* (*LSTM* adversaire), nous remarquons que le réseau adversaire est plus performant et surpasse le *LSTM* classique pour tous les traits et plus précisément pour le trait Conscientieux (CON) où la différence est marquée avec un f-score de 71% pour *LSTM-ADV* contre 43% pour le *LSTM* classique. Pour une même architecture et hyperparamètres, la génération d'exemples adversaires a permis d'améliorer les performances du *LSTM*. Aussi, le *LSTM-ADV* montre de bonnes performances pour le trait d'ouverture avec un f-score de 73% et une accuracy de 65% suivi du réseau convolutif *CNN* qui offre des performances tout aussi intéressantes pour ce trait avec 70% de f-score et 66% de accuracy. Les meilleures performances pour le trait d'extraversion (EXT) sont aussi données par le *LSTM-ADV* avec 72% de accuracy. Le réseau *Bi-RNN* quant à lui prédit avec une accuracy et f-score de 67%. Le modèle qui

produit les meilleures performances pour le trait de Névrosisme (NEU) est le *Bi-RNN* avec des performances proches du *LSTM-ADV*. Après analyse du tableau, nous observons que les modèles *LSTM-ADV*, *CNN* et *Bi-RNN* se détachent du lot avec des résultats supérieurs aux autres modèles.

5.2.2 Comparaisons

Afin d’avoir une vue d’ensemble, nous proposons de comparer nos résultats avec les performances obtenues par d’autres études ayant abordé ce problème et qui ont utilisé *myPersonality* comme corpus d’apprentissage.

- La première approche comparée est celle de Tandra [84] qui a testé plusieurs méthodes d’apprentissage profond sur le corpus *myPersonality*.
- La seconde approche est celle de Tadesse [85] qui propose une méthode différente, en procédant à une catégorisation par trait selon des dictionnaires de caractéristiques linguistiques, puis effectue une étude corrélative entre les caractéristiques et chaque trait, et procède à une extraction de caractéristiques selon les corrélations obtenues. Les caractéristiques les plus corrélées à chaque trait seront utilisées pour enfin effectuer une prédiction de traits à partir de ces dernières.

La figure 2 présente un tableau récapitulatif des précisions obtenues pour chaque trait.

Trait \ Modèle	Ouverture	Conscientieux	Extraversion	Amabilité	Névrosisme
LSTM-ADV	0.65	0.67	0.72	0.60	0.65
CNN	0.66	0.75	0.50	0.64	0.60
Bi-RNN	0.64	0.70	0.68	0.69	0.65
GRU-TANDERA	0.68	0.62	0.58	0.65	0.64
CNN-TANDERA	0.79	0.50	0.60	0.67	0.61
LSTM+CNN-TANDERA	0.75	0.57	0.71	0.50	0.58
MLP-TANDERA ¹	0.79	0.59	0.78	0.56	0.79
XGBoost-TADESSE	0.73	0.62	0.74	0.59	0.63

MLP-TANDERA : état de l’art.

5.2.3 Interpretations

En comparant les résultats de la figure suivante, on remarque que les modèles que nous proposons surpassent le meilleur modèle de Tadesse pour les traits Conscientieux (CON), Amabilité (AGR) et Névrosisme (NEU).

Aussi, les performances de nos modèles (*LSTM-ADV*, *CNN*, *Bi-RNN*) sont meilleures que l'ensemble des modèles des autres études pour le trait Conscientieux (CON), avec une accuracy de 75% pour le *CNN*, qui, à notre connaissance a atteint une performance de pointe pour ce dataset. Pour le trait d'Amabilité (AGR), le réseau *Bi-RNN* réalise de meilleures performances que les modèles *MLP-Tandera* et *XGBoost-Tadesse* avec une accuracy de 69%.

Cependant, bien que des améliorations ont été constatées, de façon générale les résultats obtenus par les modèles que nous proposons et ceux de Tandera et Tadesse restent relativement modestes. Ceci peut probablement être expliqué par la taille réduite de notre corpus d'apprentissage (10.000 statuts Facebook).

A cet effet, nous testons le réseau LSTM adversaire virtuel (*LSTM-V-ADV*) pour de l'apprentissage semi-supervisé sur un corpus résultant de l'agrégation du corpus *myPersonality* (étiqueté) et d'un corpus issu du réseau social Reddit (non-étiqueté), la taille totale du corpus obtenu est de 1 million et 10.000 statuts.

L'entraînement du *LSTM-V-ADV* a été arrêté à 5 itérations (epochs) en raison des limitations en matière de puissance de calcul auxquelles nous avons fait face. Néanmoins, les résultats obtenus sont encourageants comme l'illustre la figure suivante, représentant les courbes d'erreurs et de l'accuracy, lors de l'entraînement *LSTM-V-ADV* pour le trait CON. L'accuracy à la 5ème itération est de 65% sur l'ensemble d'apprentissage et de 62% sur l'ensemble du test. Nous pensons qu'à terme ces résultats peuvent être améliorés à condition d'entraîner le réseau plus longtemps.

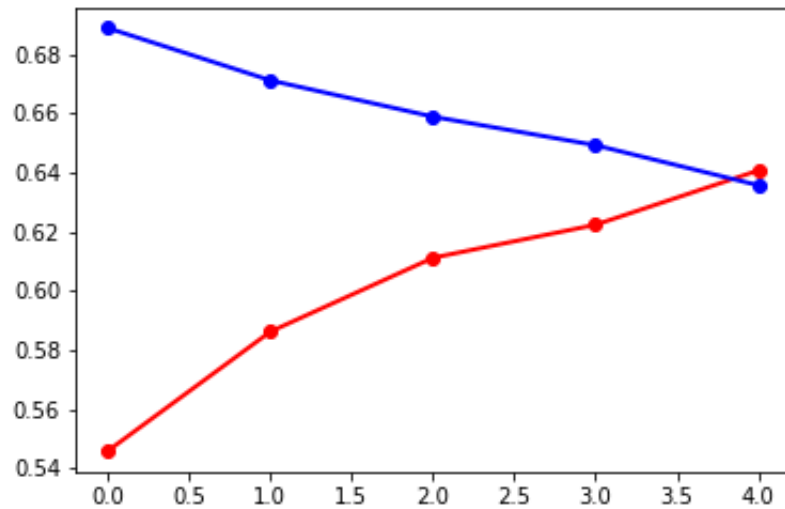


FIGURE 5.1 – Courbe d’erreur (rouge) et accuracy (bleu) du LSTM-V-ADV pour le trait CON

Enfin, nous obtenons de bonnes performances pour les traits CON et AGR, qui sont les traits les moins bien prédits par les modèles de Tadesse et Tandra.

Dans un scénario inverse, le trait OPN est le mieux prédit dans les travaux Tandra et Tadesse. Cette différence de performance peut être justifiée par les différences d’approches en premier lieu. En effet, l’approche de Tadesse repose sur une modélisation textuelle différente de celle que nous proposons. Ce contraste dans les approches peut être à l’origine des écarts reportés plus haut. Notre choix de s’appuyer sur les plongements lexicaux a permis de capturer des informations et dépendances contextuelles, cet aspect n’est pas pris en compte dans l’approche Tadesse. D’une modélisation à une autre, les caractéristiques lexicales, syntaxiques et sémantiques des textes peuvent être mises en valeur par une approche et ignorées ou moins valorisées par une autre. Aussi, le prétraitement, dans son rôle d’enrichissement ou d’omission d’informations, a pu être un facteur déterminant sur les performances obtenues. Les choix effectués tout au long de notre étude, tels que notre volonté de conserver le style d’écriture des utilisateurs, en ne procédant pas à la suppression des mots vides, en mettant en valeur les elongations et répétitions mais aussi l’expressivité émotionnelle des utilisateurs à travers la conservation des pictogrammes sous forme textuelle, ont éventuellement permis de mettre en évidence certaines caractéristiques d’écriture. Mais la conservation ou transformation de ces informations a pu avoir un effet indésirable, similaire à du bruit dans les données, empêchant une bonne prédiction pour certains traits comme constaté pour le trait OPN.

D'autres paramètres peuvent également justifier ces différences comme la dimension des plongements lexicaux (nombre de caractéristiques associées à chaque mot du vocabulaire) et leur type qu'ils soient pré-entraînés ou personnalisés, ou encore les architectures et types de modèles utilisés. On peut également citer la taille des vecteurs en entrée qui dans notre cas est égale à la médiane des longueurs des statuts, contrairement à Tandra où la taille du statut maximum est privilégiée.

Conclusion générale

Nous traitons tout au long de ce travail le problème de prédiction des traits de personnalité sur les réseaux sociaux à partir de données textuelles.

Notre premier défi fut de poser une définition au concept, souvent ambiguë, de la personnalité. Il existe de nombreuses définitions de la personnalité et presque tous les grands psychologues de la personnalité en ont proposé, toutefois la majorité s'accorde sur certains marqueurs, que nous avons détaillés dans le chapitre 1.

Une seconde difficulté durant ce travail a été de décider d'un modèle représentatif de la personnalité, afin de rendre possible l'utilisation d'instruments d'auto-évaluation. Notre choix s'est porté sur le modèle des cinq facteurs (Big Five) en raison des corrélations qui ont été démontrées entre les 5 traits de personnalité Big Five et une vaste gamme de comportements.

Afin d'étudier la relation entre l'expression textuelle d'un individu et ses traits de personnalité nous avons décidé d'exploiter les réseaux sociaux, qui représentent une vaste source de données textuelles.

Plus particulièrement, nous avons mené notre étude sur l'échantillon du corpus *myPersonality*, ensemble de statuts Facebook collectés par le centre psychométrique de l'université de Cambridge.

Afin de prédire les traits de personnalité sur le corpus *myPersonality*, nous avons proposé une approche basée sur les réseaux de neurones et le plongement lexical et avons implémenté et testé différentes méthodes notamment des modèles prouvés efficaces en classification de texte tels que *CNN*, *Bi-RNN* et *LSTM*, ainsi que des techniques d'entraînement adversaire d'ordinaire appliquées en traitement automatique de l'image, sur un réseau de type *LSTM* afin de développer un modèle plus robuste.

Nous avons également exploré la piste de l'apprentissage semi-supervisé, en expérimentant pour la première fois l'entraînement adversaire virtuel pour la tâche de prédiction des traits de personnalité.

A la suite de la phase de tests, les modèles ayant donné les meilleurs résultats sont *LSTM-ADV*, *Bi-RNN* et *CNN*. Les performances obtenues sont intéressantes notamment pour le trait Conscientieux (CON), et encourageantes pour les traits d'Ouverture (OPN) Extraversion (EXT), Amabilité (AGR) et Névrosisme (NEU). L'approche proposée semble, selon les performances obtenues lors des tests, surpasser pour une majorité de traits d'autres approches comme celles basées sur l'extraction de caractéristiques (Tadesse).

Par ailleurs, les divergences qui existent, dans les performances de prédiction des traits, entre les méthodes proposées et celles de Tadesse et Tandra, sont la conséquence de différentes techniques de prétraitement, architectures et paramètres. Nous projetons de procéder à des expérimentations plus poussées concernant les méthodes de prétraitement afin d'étudier l'impact de la suppression ou l'ajout de certains éléments tels que les mots vides ou encore les balises qui rendent compte du style d'écriture (elongation, répétition, majuscules, etc.) sur les performances de prédiction de chacun des 5 traits.

Aussi, une exploration plus approfondie des techniques d'apprentissage notamment des réseaux de neurones tel que le *CNN* avec entraînement adversaire et l'entraînement du *LSTM-V-ADV* sur un plus grand nombre d'itérations, sont des perspectives intéressantes.

De plus, la constitution d'un corpus étiqueté plus large et plus représentatif pourrait amener à de meilleurs résultats, bien que cette tâche soit difficile à mettre en place.

Enfin, le plafonnement des performances quelque soit l'approche de représentation textuelle utilisée nous amène à penser que les données textuelles des réseaux sociaux, à elles seules ne seraient pas suffisantes pour exprimer fidèlement le comportement humain. Ainsi la prise en compte et l'apport d'autres types de données tels que les images ou les données d'activité des utilisateurs (likes, partages) et autres données interactionnelles seraient précieuses.

Appendices

Application

Au travers de notre application web, nous offrons aux utilisateurs trois services distincts qui se présentent comme suit :

— **Le test de personnalité :**

Nous proposons à l'utilisateur de passer un test de personnalité standardisé et validé par des études empiriques, afin de lui permettre une meilleure compréhension de sa personne ou encore de voir comment il est perçu par les autres. A l'issue du test, le répondant peut voir ses scores ainsi qu'une interprétation de ces derniers.

— **La collecte de données :**

Une fois que l'utilisateur complète le test, nous lui proposons de nous partager ses archives de publications dans le média social de son choix. Si le participant consent à nous donner accès à ses données, nous les sauvegardons dans une base de données externe (basée documents), où chaque document est nommé selon un ID attribué de façon automatique à l'utilisateur, permettant ainsi un étiquetage de ses données textuelles. Plus précisément, la tâche de labellisation se fera à partir de la base de données où chaque couple, de score et ID de l'utilisateur, est stocké.

Cette fonctionnalité de collecte d'archives nous permettra à long terme de recueillir un corpus nécessaire à l'amélioration de nos modèles de prédictions. Il serait intéressant de cibler des internautes algériens de façon à apporter une contribution au TALN appliqué au dialecte algérien.

— **La prédiction de traits de personnalité :**

Cette partie du site est une application directe de notre projet, nous y proposons une page où l'internaute a la possibilité de nous transférer ses données textuelles, données préalablement téléchargées du réseau social de son choix, et de se voir prédire ses traits de personnalité selon le modèle des cinq facteurs (Big Five). Ce service comprend un prétraitement, une représentation du texte suivant le plongement lexical et enfin la prédiction.

A.1 Scénario d'utilisation :

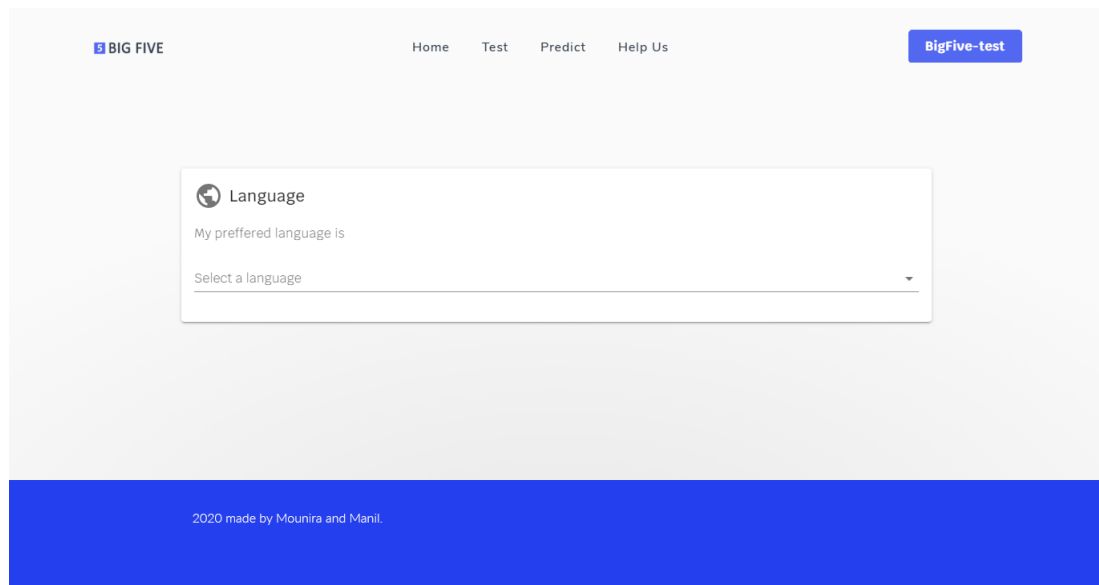
A.1.1 Passer le test de personnalité :

Via la page d'accueil représentée ci-dessous, l'utilisateur a la possibilité de choisir de passer le test de personnalité en cliquant sur le bouton **BigFive-test**.



Le test proposé dans notre application est la version de 120 questions du NEO-PR, qui dure environ 10 min. Nous avons fait le choix d'implémenter cette version du test car ce questionnaire est beaucoup plus rapide à remplir que la version originale de 300 questions tout en donnant des résultats aussi fidèles.

Comme le questionnaire a été traduit en 20 langues, et que chaque traduction a été validée par des études empiriques, nous avons décidé de permettre à nos utilisateurs de choisir la langue avec laquelle ils se sentent le plus à l'aise, dans le but d'attirer des participants d'origines diverses.



The screenshot displays the user interface of the 'Big Five' test application. At the top, a navigation bar includes the 'BIG FIVE' logo, links for 'Home', 'Test', 'Predict', and 'Help Us', and a 'BigFive-test' button. The main content area features a 'Language' selection box with a globe icon, the text 'My preferred language is', and a dropdown menu labeled 'Select a language'. The footer is a solid blue bar with the text '2020 made by Mounira and Manil'.

Parmi les 20 langues disponibles, nous retrouvons l'anglais, le français et l'arabe.

The screenshot shows the 'Big Five' personality test application. At the top, there is a navigation bar with links: 'Home', 'Test', 'Predict', and 'Help Us'. A 'BigFive-test' button is located in the top right corner. Below the navigation bar, a progress bar indicates '0%' completion. The main content area contains three personality trait sections, each with five radio button options: 'Worry about things', 'Make friends easily', and 'Have a vivid imagination'. Each section has options for 'Very Inaccurate', 'Moderately Inaccurate', 'Neither Accurate Nor Inaccurate', 'Moderately Accurate', and 'Very Accurate'. At the bottom of the form, there are 'BACK' and 'NEXT' buttons. A footer at the very bottom states '2020 made by Mounira and Manil.'

BIG FIVE Home Test Predict Help Us **BigFive-test** 0:12

0%

Worry about things

- ☐ Very Inaccurate
- ☐ Moderately Inaccurate
- ☐ Neither Accurate Nor Inaccurate
- ☐ Moderately Accurate
- ☐ Very Accurate

Make friends easily


- ☐ Very Inaccurate
- ☐ Moderately Inaccurate
- ☐ Neither Accurate Nor Inaccurate
- ☐ Moderately Accurate
- ☐ Very Accurate

Have a vivid imagination

- ☐ Very Inaccurate
- ☐ Moderately Inaccurate
- ☐ Neither Accurate Nor Inaccurate
- ☐ Moderately Accurate
- ☐ Very Accurate

BACK NEXT

2020 made by Mounira and Manil.

BIG FIVE

HomeTestPredictHelp Us

BigFive-test

0:12

2%

Je m'inquiète à propos de choses

☐

Fortement en désaccord

☐

Plutot en désaccord

☒

Ni en accord ni en désaccord

☐

Plutot en accord

☐

Fortement en accord

Je me fais des amis facilement

☐

Fortement en désaccord

☐

Plutot en désaccord

☐

Ni en accord ni en désaccord

☐

Plutot en accord

☐

Fortement en accord

J'ai une imagination débordante

☐

Fortement en désaccord

☒

Plutot en désaccord

☐

Ni en accord ni en désaccord

☐

Plutot en accord

☐

Fortement en accord

BACK

NEXT

2020 made by Mounira and Manil.

BIG FIVE
Home
Test
Predict
Help Us
BigFive-test

0:06

0%

أفلق حيال الأمور

☐ لا تنطبق عليّ إطلاقاً
☐ تنطبق عليّ قليلاً
☐ تنطبق عليّ أحياناً
☐ تنطبق عليّ كثيراً
☐ تنطبق عليّ دائماً

أصنع صداقات بسهولة

☐ لا تنطبق عليّ إطلاقاً
☐ تنطبق عليّ قليلاً
☐ تنطبق عليّ أحياناً
☐ تنطبق عليّ كثيراً
☐ تنطبق عليّ دائماً

أتمتع بقدرة تخيلية صافية

☐ لا تنطبق عليّ إطلاقاً
☐ تنطبق عليّ قليلاً
☐ تنطبق عليّ أحياناً
☐ تنطبق عليّ كثيراً
☐ تنطبق عليّ دائماً

BACK
NEXT

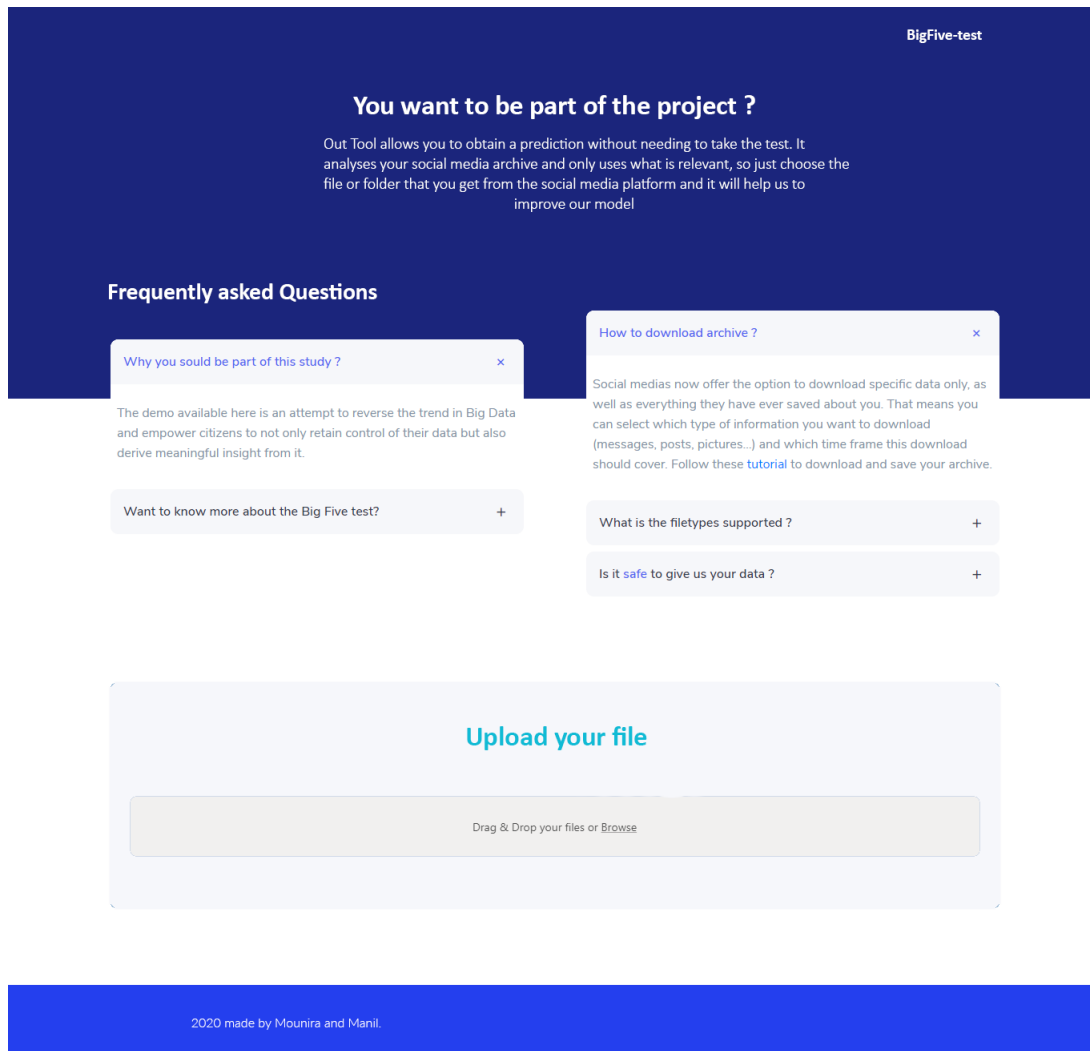
2020 made by Mounira and Manil.

A la fin du test, une interprétation des résultats est présentée à l'utilisateur afin de lui permettre de mieux comprendre ses scores. En effet à chaque degré (élevé/bas) est assimilée un trait et un ensemble de sous catégories dénommées facettes, et à chacune de ses caractéristiques psychologiques est associée une gamme de comportements.



A.2 Collecte des données :

A la fin du test, lorsque l'utilisateur voit s'afficher son score accompagné d'interprétations, nous lui proposons tout en haut de 'contribuer au projet' et cela en chargeant ses archives de publications afin de nous permettre de faire la collecte d'un corpus d'apprentissage.



A.3 Prédiction de Traits de personnalité :

Dans cette page, nous offrons à l'utilisateur une prédiction des traits de personnalité en se basant seulement sur ses données textuelles. Afin de bénéficier de ce service, il suffit simplement de choisir une archive de publications dans le réseau social de son choix. Des redirections vers des tutoriels expliquant en détail comment récupérer ses archives sont proposés à l'utilisateur.

[Home](#)
[Test](#)
[Predict](#)
[Help Us](#)

BigFive-test

See how others see you

Discover what your digital footprints reveal about your psychological profile.

Doc:

Aucun fichier choisi

Upload

Upload your archive

Our Tool allows you to obtain a prediction without needing to log in to Facebook or Twitter. It analyses your social media archive and only uses what is relevant, so just choose the file or folder that you get from the social media platform and we'll do the rest.

Facebook

Twitter

Reddit

LinkedIn

Facebook now offers the option to download specific data only, as well as everything they have ever saved about you. That means you can select which type of information you want to download (messages, posts, pictures...) and which time frame this download should cover. Follow these [tutorial](#) to download and save your facebook archive.

Frequently asked questions

How our Tool work ?

We give your archive to our artificial neural network model and we get as the output the degree (High our Low) of the 5 traits represented by the Big 5 model.

What is the filetypes supported ?

Currently supported filetypes include .txt and .csv.

Why you should try thisTool ?

The demo available here is an attempt to reverse the trend in Big Data and empower citizens to not only retain control of their data but also derive meaningful insight from it.

Is this Tool safe to use ?

Your use of this demo is completely anonymous and your results will not be stored. By using it you consent to the data submitted being used for scientific research in line with our privacy policy.

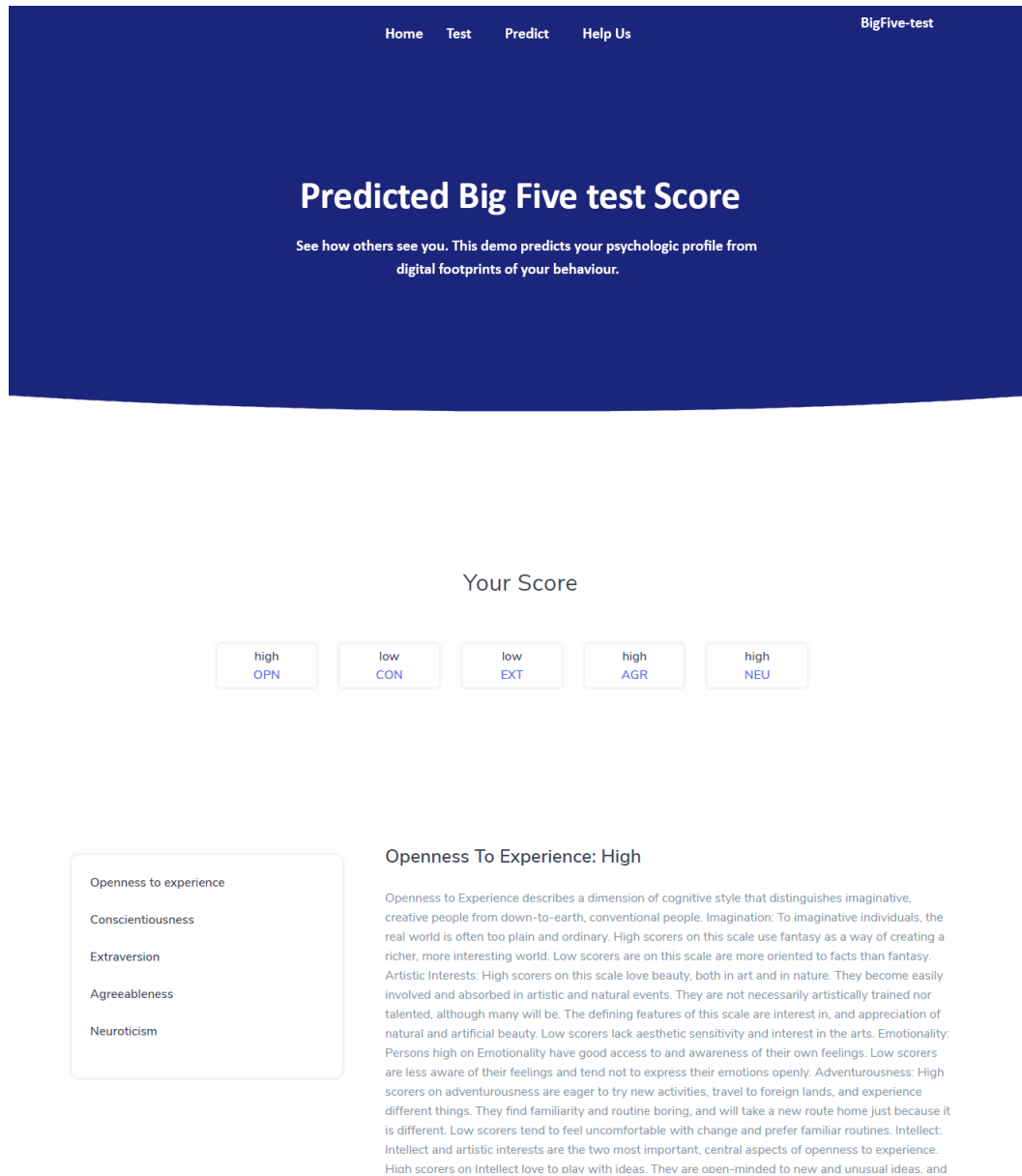
Have Question ? Get in touch!

You can take a look at our [Github](#).

Contact Us

2020 made by Mounira and Manil.

Une fois le fichier chargé en base de données une étape de prétraitement lui est appliquée. Il est ensuite passé aux 5 modèles déjà entraînés. Les résultats accompagnés de leurs interprétations sont affichés dans la page qui se charge automatiquement à la fin de la prédiction :



Covid-19

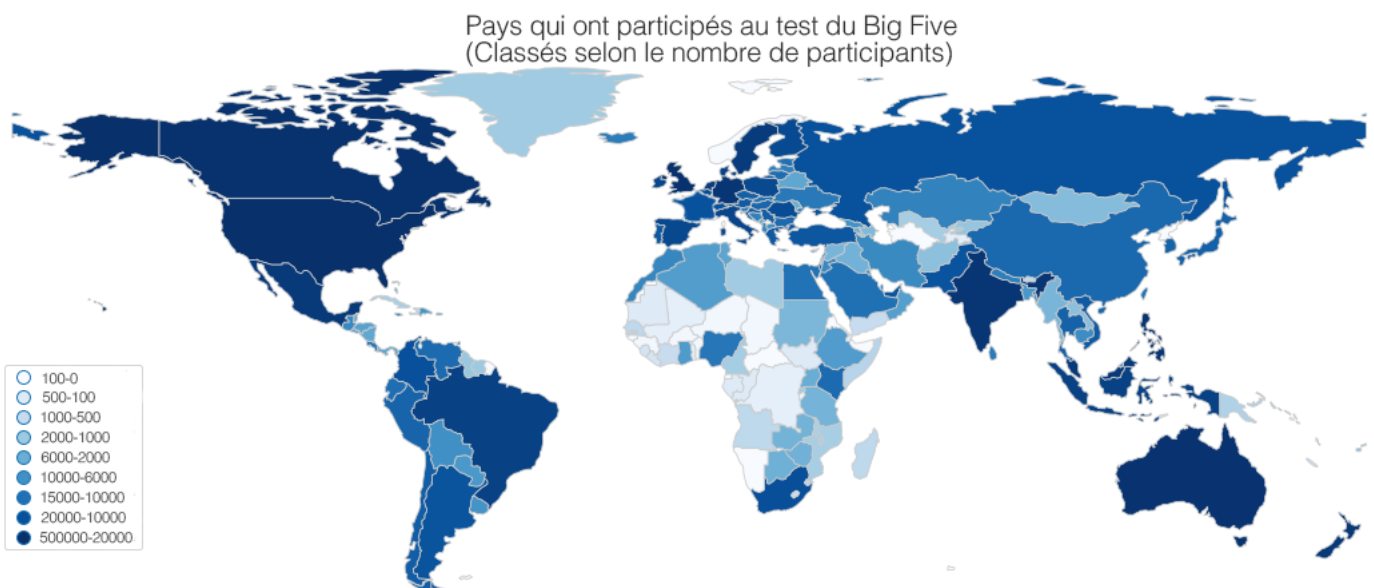
Dans ce contexte exceptionnel, où la pandémie du coronavirus bouleverse le monde que nous connaissons, nous nous sommes posé plusieurs questions autour de la thématique de notre sujet. En effet, chaque jour les médias véhiculent une grande quantité d'informations (textes, images, vidéos, analyses, etc.) sur le comportement et l'attitude des peuples de différents pays, comme par exemple : le respect ou non des consignes médicales du confinement, le comportement de solidarité ou non des membres de la société, le comportement des individus face à l'information diffusée sur le sujet, etc. Plus précisément, nous nous sommes demandé s'il était possible de trouver un quelconque lien entre les traits de la personnalité selon le modèle des cinq facteurs (Big Five) et la croissance des cas du COVID-19.

Pour répondre à cette question, nous avons exploité le dataset mis à disposition de l'université Johns Hopkins¹, qui représente la "croissance" au fil du temps des cas confirmés de coronavirus à l'échelle mondiale. Afin de comparer la propagation du virus dans chaque pays, nous avons tout d'abord calculé le ratio entre le nombre de personnes testées positives au virus et la taille de la population, et cela afin de permettre une comparaison équitable entre de pays comme les USA ou la Chine avec des nations beaucoup plus petites comme le Qatar qui, comme le montre le tableau suivant, a une densité de contaminés beaucoup plus importante.

1. Le dataset est collecté à partir de diverses sources, dont l'Organisation mondiale de la santé (OMS), BNO News, la commission nationale de la santé de la république populaire de Chine (NHC), le département de la santé de Hong Kong, le gouvernement du Canada, des Etats Unis, de Taiwan, d'Australie, le ministère de la santé de Singapour (MOH), l'European Center for Disease Prevention and Control (ECDC) et d'autres.

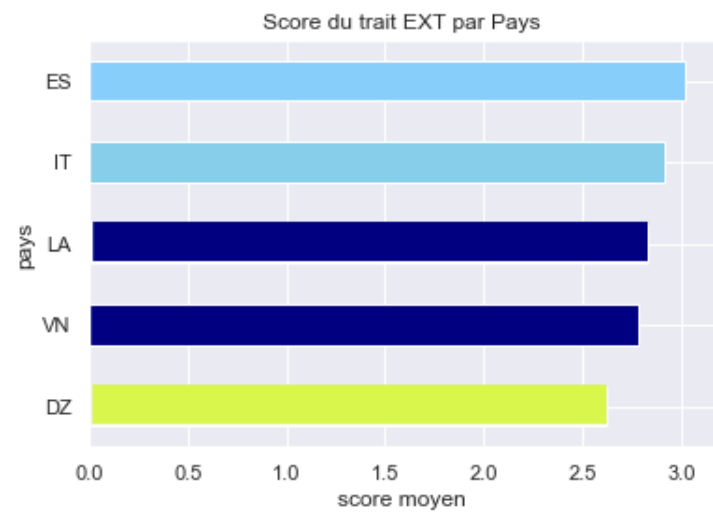
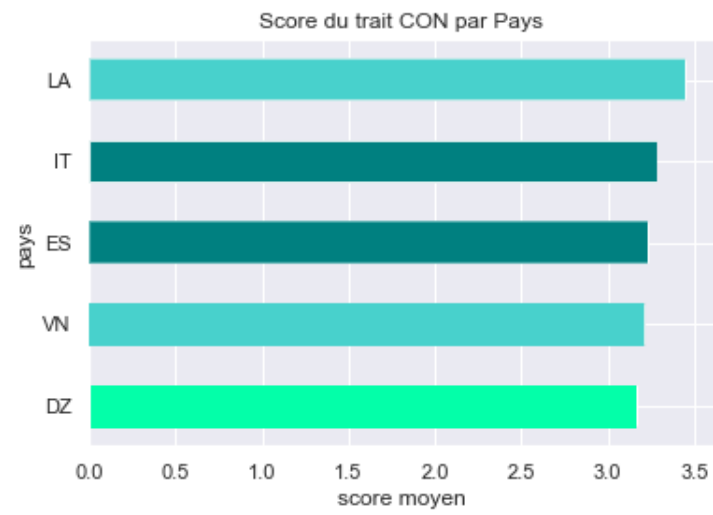
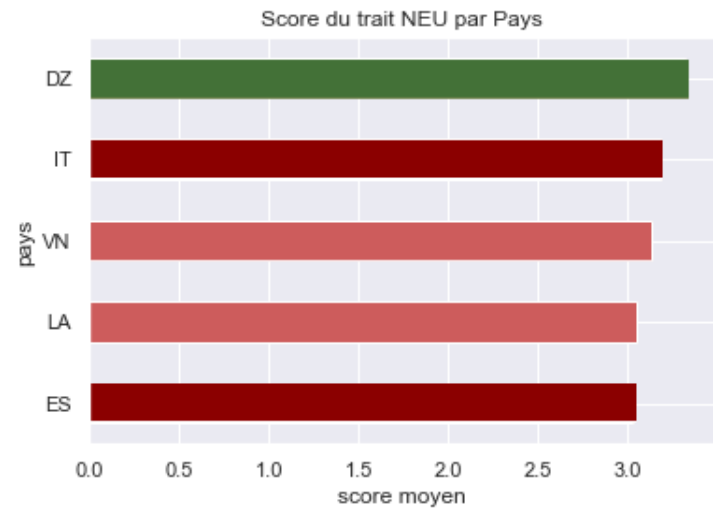
	Country	Confirmed	Population (million)	Cases per Million
0	Qatar	16191	2.83	5721.2
1	Spain	218011	46.74	4664.33
2	Ireland	21772	4.88	4461.48
3	Belgium	50267	11.54	4355.89
4	United States	1.18106e+06	329.06	3589.2
5	Italy	211938	60.55	3500.21
6	Switzerland	29981	8.59	3490.22
7	Singapore	18778	5.8	3237.59
8	United Kingdom	196852	67.53	2915.03
9	France	169588	65.13	2603.84
90	Algeria	11147	43.05	258.93
151	Vietnam	334	96.46	3.45
152	Laos	19	7.17	2.65

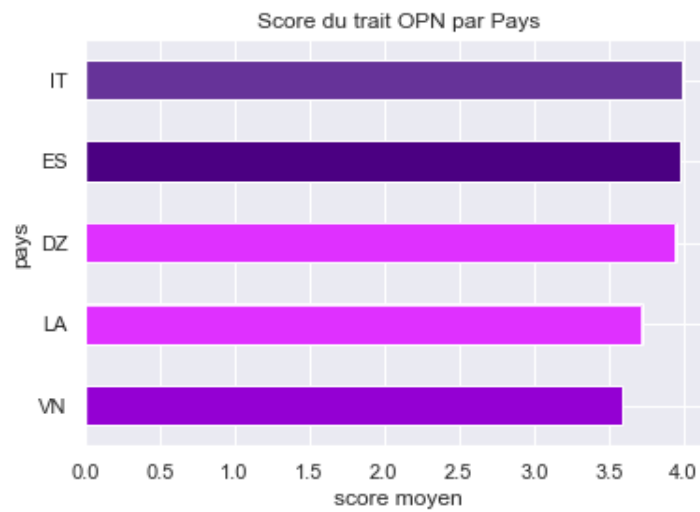
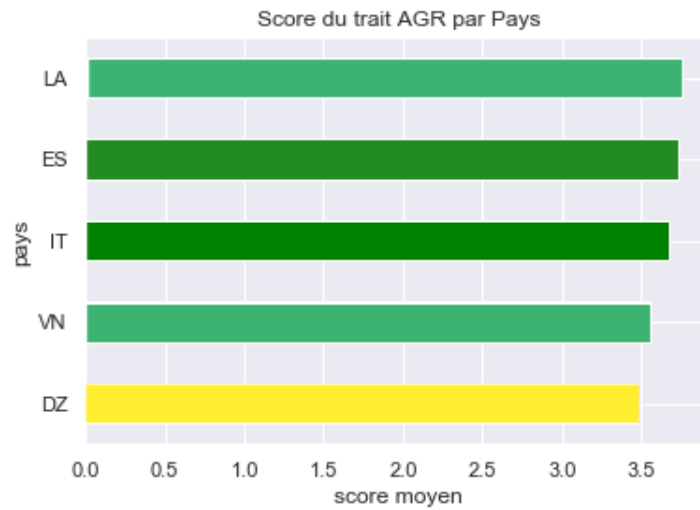
Ensuite, nous exploitons l'ensemble de données du projet *Open Source Psychometrics*², qui regroupe les scores du test de personnalité implémentant le modèle du Big Five, de 1 million de participants dans 223 pays différents, comme le montre la figure suivante.



A l'aide du dataset rassemblant les scores Big Five, nous calculons le score moyen de chaque nation suivant les 5 traits. Nous effectuons notre comparaison sur un échantillon de 5 pays qui sont : L'Espagne et l'Italie, représentant les pays à fort taux de contamination, le Laos et le Vietnam qui eux représentent les pays à faible taux de contamination et enfin l'Algérie pays qui se situe entre les deux tendances.

2. Open Source Psychometrics : est un site web qui fait passer le test du modèle des 5 facteurs





Code	Pays
ES	Espagne
IT	Italie
DZ	Algérie
LA	Laos
VN	Vietnam

Code	Trait du Big Five
OPN	Ouverture à l'expérience
CON	Consciencieux
EXT	Extraversion
AGR	Agréabilité
NEU	Névrosisme

A partir des graphes ci-dessous, qui représentent les scores moyens des traits du modèle Big Five de notre échantillon de 5 pays, nous remarquons que pour le trait ‘OPN’ (l’Ouverture à l’expérience), le score augmente avec le taux de contamination. En effet, pour l’Italie et l’Espagne qui connaissent une forte densité de personnes contaminées le score du trait OPN atteint presque la valeur 4, l’Algérie avec un taux plus faible de contamination est très légèrement en dessous de ce score et enfin les deux pays à très faible taux de contaminées selon la taille de la population (Vietnam et Laos) ont un score du trait ‘OPN’ se rapprochant plutôt vers 3.5. Ces constatations nous amènent à penser que il y aurait éventuellement une corrélation entre un trait de personnalité et l’évolution des cas de contaminations.

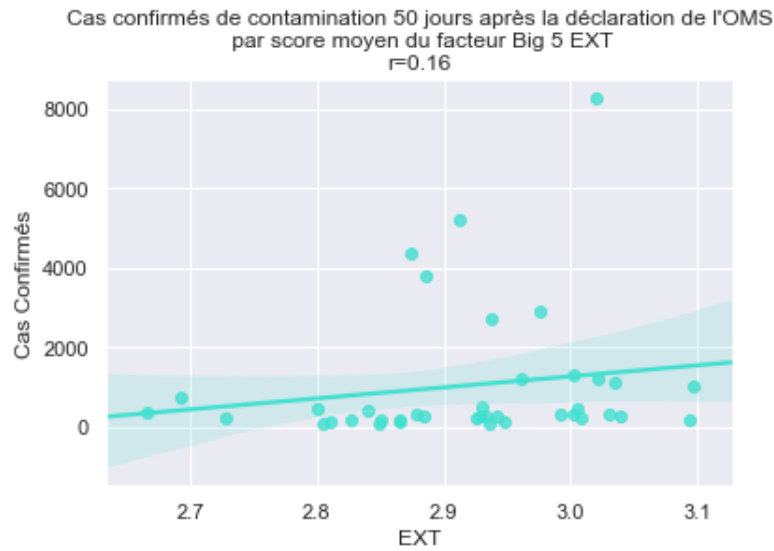
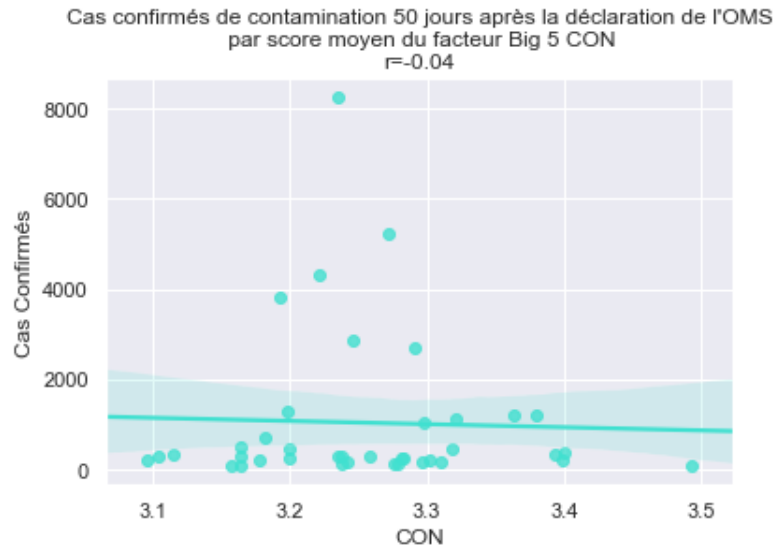
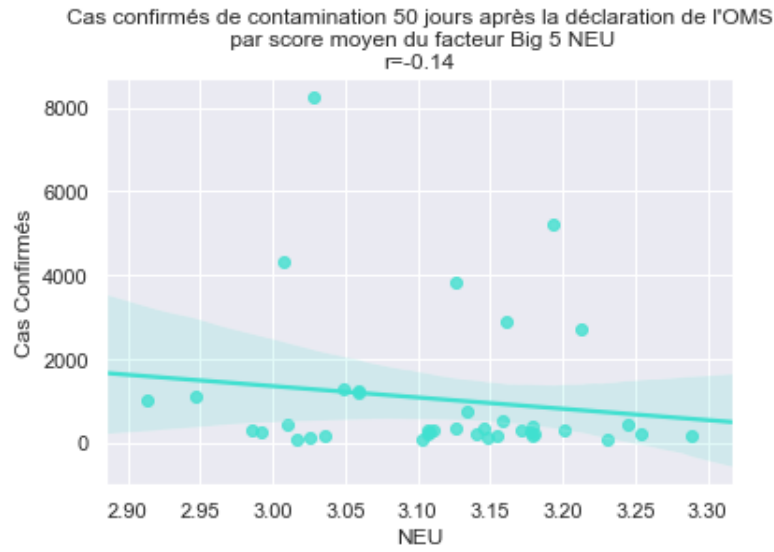
Afin de confirmer cette intuition, nous décidons d’élargir notre échantillon et de faire une étude sur la totalité des pays représentés par le dataset des scores Big Five. Comme notre but est d’étudier le comportement des individus indépendamment des ressources sanitaires dont disposent leurs pays, nous avons choisi de prendre en compte l’évolution des cas de contamination confirmés au coronavirus allant du 11 Mars 2020, la date officielle ou l’OMS³ a déclarée le virus comme pandémie et a mis en garde les populations de sa dangerosité et des mesures que chacun se doit de prendre pour en limiter la propagation.

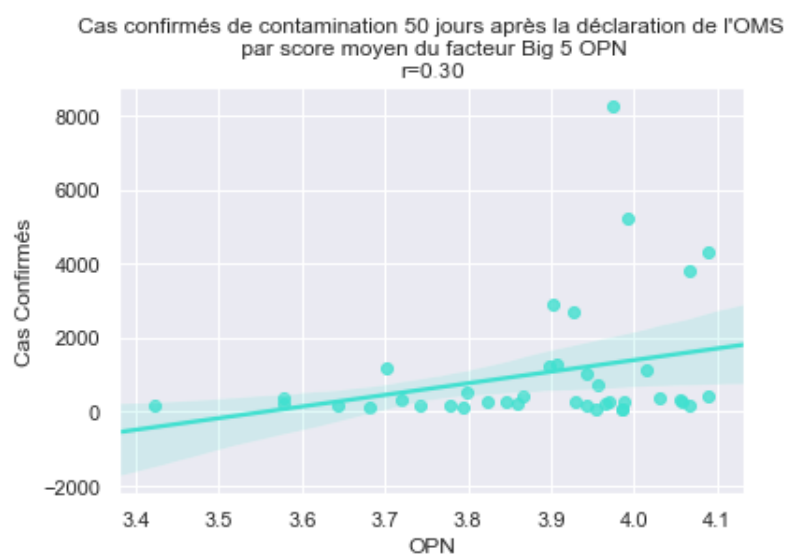
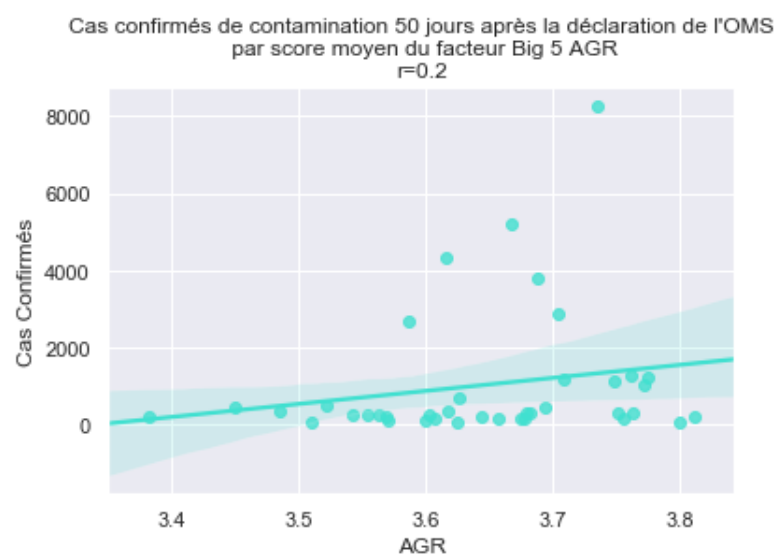
Nous étudions donc l’évolution des cas confirmés de COVID-19 allant du 11 Mars 2020 jusqu’à une période de 50 jours, la durée optimale estimée par [86], afin d’observer d’éventuelles améliorations après l’instauration de mesures de restrictions sanitaires. Cette étape est primordiale pour ne pas fausser la comparaison, car nous voulons ici étudier le comportement de l’individu et non les ressources des pays, nous supposons donc les populations sensibilisées dès lors que l’OMS a officiellement reconnue cette catastrophe sanitaire et expliqué comment limiter sa propagation.

Ensuite, à l’aide de l’ensemble de données du projet *Open Source Psychometrics*, nous calculons le score moyen de chaque nation en n’incluant que les pays avec au moins 1000 observations, afin que les scores soient représentatifs. Nous obtenons donc un total de 58 pays.

Enfin, nous examinons la corrélation entre les scores moyens des pays sur chacun des traits de personnalité du Big Five et les cas de contamination confirmés sur la durée s’étalant de la déclaration faite par l’OMS et 50 jours après.

3. OMS : Organisation mondiale de la santé.





Comme nous le constatons à travers les figures présentées, le seul facteur Big 5 qui semble montrer une **corrélation** avec la **croissance des cas du COVID-19** est « **OPN** » ou l'**Ouverture à l'expérience** avec un coefficient de corrélation de **Pearson** égale à **0.30**.

L'ouverture représente les capacités d'imagination, de créativité et de curiosité de la personnalité. Un individu avec un score élevé, a tendance à faire preuve d'innovation, se montre curieux et est ouvert au changement. Il est plus susceptible de s'ennuyer rapidement en cas de tâches répétitives et a tendance à rejeter les règles et les traditions. Les pays avec les scores « d'ouverture » les plus élevés ont connu une croissance plus importante au cours des 50 jours qui ont suivi la déclaration de l'OMS, comme nous le confirme le tableau suivant.

	OPN	Country	ConfirmedCases
0	4.08	Germany	4337.0
1	4.066059	France	3809.0
2	4.015059	Switzerland	1117.0
3	4.006961	Ukraine	311.0
4	3.991518	Italy	5210.0
19	3.958159	Algeria	95.0

Glossaire

analyses factorielles méthodes mathématiques expérimentales, spécialement employées en psychologie, qui ont pour objet l'étude des dimensions, ou facteurs, d'un domaine empirique donné. 13

microblogging service en ligne de textes courts. 22

psychanalytique qui relève de la psychanalyse, thérapie psychique par exploration de l'inconscient du patient. 11

psychométrie science qui mesure par des tests des caractéristiques psychologiques d'un individu. 11

taxonomie science des classifications. 8

échelle de Likert outil psychométrique permettant de mesurer une attitude chez des individus. 15

Bibliographie

- [1] Timothy A. ALLEN et Colin G. DEYOUNG., « Personality Neuroscience and the Five-Factor Model(2016) ».
- [2] Isabel Briggs with Peter B. Myers MYERS, « Understanding Personality Type. », 1995.
- [3] L. R. GOLDBERG, « An alternative “description of personality” : The big-five factor structure. », *in : Journal of Personality and Social Psychology* (1990).
- [4] J. ; Vazire S. ; Gaddis S. ; Schmukle S. ; Egloff B. ; Gosling S. BACK M. ; Stopfer, *Facebook Profiles Reflect Actual Personality, Not Self-Idealization*. 2010.
- [5] Donnellan M. B. MCADAMS K. K., « Facets of personality and drinking in first-year college students. Personality and Individual Differences », *in : Aujourd'hui la théologie* ((2009)).
- [6] Manuel IBÁÑEZ et al., « Alcohol Expectancies Mediate and Moderate the Associations between Big Five Personality Traits and Adolescent Alcohol Consumption and Alcohol-Related Problems », *in : Frontiers in Psychology* 6 (déc. 2015), DOI : 10.3389/fpsyg.2015.01838.
- [7] Sampo PAUNONEN et Michael ASHTON, « Big Five Factors and Faces and the Prediction of Behavior », *in : Journal of Personality and Social Psychology* 81 (oct. 2001), p. 524-539, DOI : 10.1037//0022-3514.81.3.524.
- [8] Ryan RHODES et N SMITH, « Personality correlates of physical activity : A review and meta-analysis », *in : British journal of sports medicine* 40 (jan. 2007), p. 958-65, DOI : 10.1136/bjbm.2006.028860.
- [9] Angelina SUTIN et al., « Personality and metabolic syndrome », *in : Age (Dordrecht, Netherlands)* 32 (déc. 2010), p. 513-9, DOI : 10.1007/s11357-010-9153-9.
- [10] Leslie MARTIN, Howard FRIEDMAN et Joseph SCHWARTZ, « Personality and Mortality Risk Across the Life Span : The Importance of Conscientiousness as a Biopsychosocial Attribute », *in : Health psychology : official journal of the Division of Health Psychology, American Psychological Association* 26 (août 2007), p. 428-36, DOI : 10.1037/0278-6133.26.4.428.
- [11] Brent ROBERTS et al., « The Power of Personality The Comparative Validity of Personality Traits, Socioeconomic Status, and Cognitive Ability for Predicting Important Life Outcomes », *in : Perspectives on Psychological Science* 2 (déc. 2007), DOI : 10.1111/j.1745-6916.2007.00047.x.

- [12] Friedman HS. GOODWIN RD, « Health status and the five-factor personality traits in a nationally representative sample. », *in* : *J Health Psychol.* 2 (déc. 2006), DOI : 0.1177/1359105306066610.
- [13] Daniel OZER et Veronica BENET, « Personality and the Prediction of Consequential Outcomes », *in* : *Annual review of psychology* 57 (fév. 2006), p. 401-21, DOI : 10.1146/annurev.psych.57.102904.190127.
- [14] Ian J. GOODFELLOW, Jonathon SHLENS et Christian SZEGEDY, *Explaining and Harnessing Adversarial Examples*, 2014, arXiv : 1412.6572 [stat.ML].
- [15] Robert M. SACCUZZO Dennis P.; Kaplan, *Psychological Testing : Principles, Applications, and Issues (7th ed.)*. Wadsworth Cengage Learning.
- [16] S. FREUD, « The interpretation of dreams. Basic Books. », *in* : 1955.
- [17] B. F. SKINNER, *The behavior of organisms : an experimental analysis*. Appleton-Century. 1938.
- [18] Costa P. T. McCRAE R. R., *Validation of the five-factor model of personality across instruments and observers*. 1987.
- [19] Matthias BURISCH, « Approaches to personality inventory construction : A comparison of merits », *in* : *American Psychologist* 39 (mar. 1984), p. 214-227, DOI : 10.1037/0003-066X.39.3.214.
- [20] H. S. ALLPORT G. W. Odbert, *Trait names : A psycholexical study*. 1936.
- [21] W. McDOUGALL, « Of the words character and personality. Character Personality », *in* : *A Quarterly for Psychodiagnostic Allied Studies* (mar. 1932), p. 214-227.
- [22] Raymond B CATTELL, Mary D CATTELL et Edgar JOHNS, *High school personality questionnaire*, Institute for Personality et Ability Testing, Incorporated, 1984.
- [23] W. T. NORMAN, *Toward an adequate taxonomy of personality attributes : replicated factor structure in peer nomination personality ratings*. 1963.
- [24] Timothy ALLEN et Colin DEYOUNG, « Personality Neuroscience and the Five-Factor Model », *in* : jan. 2016, DOI : 10.1093/oxfordhb/9780199352487.013.26.
- [25] R.R COSTA P.T. et McCrae, *Revised NEO Personality Inventory (NEO-PI-R) and NEO Five-Factor Inventory (NEO-FFI) manual*, Psychological Assessment Resources, 1992.
- [26] Greene R. L. year = 2008 month = 01 pages = 105-125 title = Handbook of personality assessment. John Wiley Sons Inc. WEINER I. B., *in* :
- [27] Robert McCRAE, « The Five-Factor Model of Personality Across Cultures », *in* : jan. 2002, p. 105-125, ISBN : 978-0-306-47355-5, DOI : 10.1007/978-1-4615-0763-5_6.

- [28] C. G. JUNG, *Collected Works of C.G. Jung*.
- [29] I. B. MYERS, « The Myers-Briggs Type Indicator : Manual », *in* : 1962.
- [30] Isabel Briggs MYERS et al., *MBTI manual : A guide to the development and use of the Myers-Briggs Type Indicator*, t. 3, Consulting Psychologists Press Palo Alto, CA, 1998.
- [31] Thomas G LONG, « Myers-Briggs et autres modernes Astrologies », *in* : *Aujourd'hui la théologie* ((2016)).
- [32] Robert PLOYHART et P. BLIESE, « Individual Adaptability (I-ADAPT) Theory : Conceptualizing the Antecedents, Consequences, and Measurement of Individual Differences in Adaptability », *in* : t. 6, jan. 2006, p. 3-39, ISBN : 978-0-7623-1248-1, DOI : 10.1016/S1479-3601(05)06001-7.
- [33] Stephen J. DOLLINGER, « Research Note : Personality and Music Preference : Extraversion and Excitement Seeking or Openness to Experience ? », *in* : *Psychology of Music* 21.1 (1993), p. 73-77.
- [34] John T. JOST, Tessa V. WEST et Samuel D. GOSLING, « Personality and ideology as determinants of candidate preferences and “Obama conversion” in the 2008 U.S. presidential election », *in* : *Du Bois Review : Social Science Research on Race* 6.1 (2009), 103–124.
- [35] James W PENNEBAKER et KING, *Linguistic styles : Language use as an individual difference. Journal of personality and social psychology*. 1999, arXiv : 1412.6572 [stat.ML].
- [36] Donahue E. M. Kentle R. L. JOHN O. P., « The Big Five Inventory–Versions 4a and 54. », *in* : *Institute of Personality and Social Research* 28 (1991).
- [37] « Reddit : A Gold Mine for Personality Prediction ».
- [38] Daelemans W. Plank B. VERHOEVEN B., « TwiSty : A Multilingual Twitter Stylometry Corpus for Gender and Personality Profiling ».
- [39] M. Sidnal. KALGHATGI M.P. ; Ramannavar, « A neural network approach to personality prediction based on the bigfive model », *in* : *Int. J. Innov. Res. Adv. Eng.* 2 (2014).
- [40] N. MAJUMDER et al., « Deep Learning-Based Document Modeling for Personality Detection from Text », *in* : *IEEE Intelligent Systems* 32.2 (2017), p. 74-79.
- [41] Francois MAIRESSE et al., « Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text. », *in* : *J. Artif. Intell. Res. (JAIR)* 30 (sept. 2007), p. 457-500.
- [42] Walter LUYCKX Kim Daelemans, « Personae : a Corpus for Author and Personality Prediction from Text. »

- [43] Guan Z. Hao B. Bai S. NIE D. et ZHU, *Predicting personality on social media with semi-supervised learning. Proceedings*, 2014, arXiv : 1412.6572 [stat.ML].
- [44] Viktor HANGYA et al., « SZTE-NLP : Aspect level opinion mining exploiting syntactic cues », *in : SemEval@COLING*, 2014.
- [45] Lingpeng KONG et al., « A Dependency Parser for Tweets », *in : Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar : Association for Computational Linguistics, 2014.
- [46] Lehireche.R HAMOU.M., *Representation of textual documents by the approach word-net and n-grams for the unsupervised classification (clustering) with 2D cellular automata : a comparative study*. 2009.
- [47] S. Yan J. Yan K. Xie W. PU N. Liu et Z. CHEN., *Local word bag model for text categorization*. 2007.
- [48] Taketoshi Yoshida WEN ZHANG et Xijin TANG., *A comparative study of tf*idf, lsi and multi-words for text classification. Expert Systems with Applications*, 2011.
- [49] Z. HARRIS, « Distributional structure. », *in* : 1954.
- [50] Tomas MIKOLOV et al., *Efficient Estimation of Word Representations in Vector Space*, 2013, arXiv : 1301.3781 [cs.CL].
- [51] « http://mediamining.univ-lyon2.fr/people/guille/word_embedding/cbow.html ».
- [52] Lilian WENG, « <https://lilianweng.github.io/lil-log/2017/10/15/learning-word-embedding.html> ».
- [53] Jeffrey PENNINGTON, Richard SOCHER et Christopher MANNING, « Glove : Global Vectors for Word Representation », *in : Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar : Association for Computational Linguistics, oct. 2014, p. 1532-1543, DOI : 10.3115/v1/D14-1162, URL : <https://www.aclweb.org/anthology/D14-1162>.
- [54] « http://mediamining.univ-lyon2.fr/people/guille/word_embedding/glove.html ».
- [55] BONASTRE, « ApprentissageAutoIntroductionI ».
- [56] Fabien MOUTARDE, « Cours Apprentissage MinesParisTech », 2011.
- [57] Alexander Zien OLIVIER CHAPELLE Bernhard Scholkopf, *Semi-Supervised Learning*, MITPress.
- [58] Mark SCHMIDT, « EM for Semi-Supervised Learning and Mixture Models ».
- [59] « <https://www.irif.fr/~kesner/enseignement/iup/cours81.pdf> ».
- [60] « <https://www.labri.fr/perso/nrougier/downloads/Perceptron.pdf> ».
- [61] « <http://www.isir.upmc.fr/UserFiles/File/LPrevost/connex%201%20%202.pdf> ».
- [62] « <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networksoverview> ».

- [63] « <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> ».
- [64] « <https://stanford.edu/shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks> ».
- [65] Kui REN et al., « Adversarial Attacks and Defenses in Deep Learning », in : *Engineering* 6.3 (2020), p. 346 -360, ISSN : 2095-8099, DOI : <https://doi.org/10.1016/j.eng.2019.12.012>, URL : <http://www.sciencedirect.com/science/article/pii/S209580991930503X>.
- [66] Xiaoyong YUAN et al., *Adversarial Examples : Attacks and Defenses for Deep Learning*, 2017, arXiv : 1712.07107 [cs.LG].
- [67] « https://adversarial-ml-tutorial.org/adversarial_training/ ».
- [68] Takeru MIYATO, Andrew M. DAI et Ian GOODFELLOW, *Adversarial Training Methods for Semi-Supervised Text Classification*, 2016, arXiv : 1605.07725 [stat.ML].
- [69] Takeru MIYATO et al., « Virtual Adversarial Training : A Regularization Method for Supervised and Semi-Supervised Learning », in : *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP (avr. 2017), DOI : 10.1109/TPAMI.2018.2858821.
- [70] Gerard. SALTON, *Automatic Information Organization and Retrieval*. McGraw Hill Text, 1968, ISBN : 0070544859.
- [71] Srivastava S. GOSLING SD Vazire S, « Should we trust web-based studies ? A comparative analysis of six preconceptions about internet questionnaires. », in : *Am Psychol* (2004), p. 93-104, DOI : 10.1037/0003-066X.59.2.93.
- [72] Mike Swarbrick Jones. Evolution AI, « The Reddit Self-Post Classification Task (RSPCT) : a highly multiclass dataset for text classification. »
- [73] Jean-Marc DEWAELE et Adrian FURNHAM, « Personality and speech production : A pilot study of second language learners », in : *Personality and Individual Differences* 28 (fév. 2000), p. 355-365, DOI : 10.1016/S0191-8869(99)00106-3.
- [74] Thomas HOLTGRAVES, « Social Psychology and Language : Words, Utterances, and Conversations », in : juin 2010, ISBN : 9780470561119, DOI : 10.1002/9780470561119.socpsy002036.
- [75] Matthias MEHL, Samuel GOSLING et James PENNEBAKER, « Personality in its natural habitat : Manifestations and implicit folk theories of personality in daily life », in : *Journal of personality and social psychology* 90 (juin 2006), p. 862-77, DOI : 10.1037/0022-3514.90.5.862.

- [76] Jeffrey PENNINGTON, Richard SOCHER et Christopher D. MANNING, « GloVe : Global Vectors for Word Representation », *in* : *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, p. 1532-1543, URL : <http://www.aclweb.org/anthology/D14-1162>.
- [77] « A systematic comparison of context-counting vs. context-predicting semantic vectors. », *in* : (), URL : <http://clic.cimec.unitn.it/marco/publications/acl2014/baronietal-countpredict-acl2014.pdf>.
- [78] Thomas K. LANDAUER et Susan T. DUMAIS, « A solution to Plato's problem : The latent semantic analysis theory of acquisition, induction, and representation of knowledge », *in* : 1997.
- [79] Kevin Jamieson Giulia DeSalvo Afshin Rostamizadeh Li Lisha et Ameet TALWALKAR., *Hyperband : A novel bandit-based approach to hyperparameter optimization*. 2016.
- [80] David M. BEAZLEY et Brian K. JONES, *Python cookbook*, 3^a édition, Sebastopol : O'Reilly, 2013.
- [81] Steven BIRD, Ewan KLEIN et Edward LOPER, *Natural Language Processing with Python*, 1st, O'Reilly Media, Inc., 2009, ISBN : 0596516495.
- [82] Bharath RAMSUNDAR et Reza Bosagh ZADEH, *TensorFlow for Deep Learning : From Linear Regression to Reinforcement Learning*, 1st, O'Reilly Media, Inc., 2018, ISBN : 1491980451.
- [83] Antonio GULLI et Sujit PAL, *Deep Learning with Keras*, Packt Publishing, 2017, ISBN : 1787128423.
- [84] Hendro Suhartono D. Wongso R. Prasetyo Y. L. TANDERA T., *Personality Prediction System from Facebook Users. In Procedia Computer Science*. 2017.
- [85] M. M. TADESSE et al., « Personality Predictions Based on User Behavior on the Facebook Social Media Platform », *in* : *IEEE Access* 6 (2018), p. 61959-61969.
- [86] Fernando ALVAREZ, David ARGENTE et Francesco LIPPI, « A Simple Planning Problem for COVID-19 Lockdown », *in* : *SSRN Electronic Journal* (jan. 2020), DOI : 10.2139/ssrn.3569911.
- [87] Daniele QUERCIA et al., « Our Twitter Profiles, Our Selves : Predicting Personality with Twitter », *in* : oct. 2011, p. 180-185.
- [88] « Personality Traits on Twitter—or—How to Get 1,500 Personality Tests in a Week ».
- [89] « How Universal Is the Big Five? Testing the Five-Factor Model of Personality Variation Among Forager–Farmers in the Bolivian Amazon 2013 ».

- [90] MURRAY R. BARRICK et MICHAEL K. MOUNT, « The big five personality dimensions and job performance : A META-ANALYSIS », *in* : *Personnel Psychology* 44.1 (1991), p. 1-26.
- [91] Lisa SAULSMAN et Andrew PAGE, « The Five-Factor Model and personality disorder empirical literature », *in* : *Clinical psychology review* 23 (fév. 2004), p. 1055-85.
- [92] « <http://pageperso.lif.univ-mrs.fr/francois.denis/IAAM1/cours-RN.pdf> ».
- [93] « [http://eric.univ-lyon2.fr/ricco/cours/slides/gradient_{descent}.pdf](http://eric.univ-lyon2.fr/ricco/cours/slides/gradient_descent.pdf) ».
- [94] Zichao YANG et al., « Hierarchical Attention Networks for Document Classification », *in* : *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, San Diego, California : Association for Computational Linguistics, juin 2016, p. 1480-1489, DOI : 10.18653/v1/N16-1174, URL : <https://www.aclweb.org/anthology/N16-1174>.
- [95] C.D. Manning H. SCHÜTZE., « Foundations of Statistical Natural Language Processing. year =1999 ».
- [96] Kim LUYCKX et Walter DAELEMANS, « Authorship Attribution and Verification with Many Authors and Limited Data », *in* : *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK : Coling 2008 Organizing Committee, août 2008, p. 513-520, URL : <https://www.aclweb.org/anthology/C08-1065>.
- [97] Hannes GRASSEGGER et Mikael KROGERUS, « The data that turned the world upside down », *in* : *Vice Motherboard* 28 (2017).

Table des figures

2.1	Les quatre fonctions de Jung	16
2.2	Les caractéristiques des 16 types (MBTI)	17
3.1	Comparaison des représentations spatiales	29
3.2	Calcul d'un plongement lexical	32
3.3	Architecture du modèle Skip-gram[52]	33
3.4	Architecture du modèle CBOW[52]	35
3.5	Pour $k=6$, l'entrée (en rouge) sera affecté à la classe A	40
3.6	L'hyperplan optimal maximise la marge, les points sur les lignes en pointillés sont appelés supports.	40
3.7	Exemple d'itérations de l'algorithme EM	43
3.8	Perceptron	44
3.9	Perceptron multicouches	47
3.10	Architecture d'un réseau de neurones récurrents	50
3.11	Architecture d'une cellule RNN[62]	50
3.12	Architecture interne d'une cellule LSTM	52
3.13	Architecture interne d'une cellule GRU	53
3.14	Architecture générale d'un réseau récurrent bidirectionnel	54
3.15	Opérations de convolution	55
3.16	Opération de pooling	56
3.17	Exemple de génération d'exemple adversaire	57
3.18	Architecture du réseau adversaire	61
4.1	Schéma général du système de prédiction de personnalité	65
4.2	Distribution des utilisateurs (selon le genre et l'âge) dans le corpus myPersonality	67
4.3	Distribution des classes (scores) pour chaque statut	68
4.4	Schéma de l'architecture pour les réseaux récurrents	75
4.5	Schéma de l'architecture du réseau CNN	79
5.1	Courbe d'erreur (rouge) et accuracy (bleu) du LSTM-V-ADV pour le trait CON	85