



République Algérienne Démocratique et Populaire



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari Boumediene

**Faculté d'Electronique et d'Informatique
Département Informatique**

Mémoire de Licence

Filière : Informatique

Spécialité : ACAD

Thème

Proposition d'une approche basée sur les trigrammes et le Word-embedding pour
l'analyse des sentiments sur Twitter

Sujet Proposé par :

M. AHSENE DJABALLAH Kamel

M. BOUKHALFA Kamel

Présenté par :

MOUFFOK Tayeb Abderraouf

NOUAR Sofia

Devant le jury composé de :

M. Président (e)

M. Membre

Binôme n° : 069 / 2020

TABLE DES MATIÈRES

CHAPITRE 1 : GÉNÉRALITÉS ET DÉFINITIONS

PARTIE 1 : GÉNÉRALITÉS SUR LES RESEAUX SOCIAUX	1
1.1. Web 2.0	2
1.2. Réseaux sociaux	2
1.2.1. Twitter	3
1.2.2. Facebook	3
1.2.3. Instagram	3
1.2.4. LinkedIn	3
PARTIE 2 : DÉFINITIONS	4
2.1. Définition d'une émotion	4
2.2. Définition d'un sentiment	4
2.3. Définition d'une opinion	4
2.4. Définition de la polarité d'une opinion	4
2.5. Définition d'une expression subjective	4
2.6. Définition de l'analyse des sentiments	5
2.7. Définition de la radicalité	5

CHAPITRE 2 : ÉTAT DE L'ART

PARTIE 1 : L'ANALYSE DES SENTIMENTS	7
1.1. Les niveaux de l'analyse des sentiments	7
1.1.1. Niveau document	7
1.1.2. Niveau phrase	7
1.1.3. Niveau entité et aspect	7
1.2. Le processus de l'analyse des sentiments	8
1.2.1. Acquisition et prétraitement du corpus	8
1.2.2. La pertinence par rapport au sujet	8
1.2.3. La détection d'opinions	8
1.2.4. Réordonnement des résultats	8
1.3. Les domaines d'application de l'analyse des sentiments	9

1.3.1. Politique	9
1.3.2. Marketing	9
1.3.3. Cyberharcèlement	9
1.3.4. Sécurité publique	10
1.4. Les problèmes de l'analyse des sentiments	10
1.4.1. Problèmes liés au contexte	10
1.4.2. Problèmes liés à la structure du texte	10
1.4.3. Problèmes liés aux fautes lexicales	11
PARTIE 2 : ETAT DE L'ART	11
2.1. LES DIFFÉRENTES APPROCHES DE L'ANALYSE DES SENTIMENTS	11
2.1.1. Approche basée sur le lexique	11
2.1.2. Approche basée apprentissage automatique	12
2.1.2.1. Supervisée	12
2.1.2.2. Non supervisée	12
2.1.3. Approche hybride	12
2.2. LE WORD EMBEDDING DANS L'ANALYSE DES SENTIMENTS	13
2.2.1. Le Word2Vec	13
2.2.1.1. CBOW	13
2.2.1.2. Skip-Gram	14
2.5. LES TRAVAUX EXISTANTS	15
2.5.1. Conde-Céspedes et al. [30]	15
2.5.2. Asif et Ishtiaq [31]	16
2.5.3. Sohail et Qadoos [32]	16
2.5.4. Rattrout et Ateeq [33]	17
2.5.5. Jamilah Rabeh Alharbi et al. [42]	17
2.5.6. Yun Wang et al. [54]	18
2.5.7. D'autres travaux dans le domaine du terrorisme	18
SYNTHÈSE	19

CHAPITRE 3 : CONCEPTION DE L'APPROCHE

1. Choix du réseau social	22
2. Elaboration et préparation des outils d'analyse	22
2.1. Conception du modèle Word2vec	22
2.1.1. Création du corpus d'apprentissage	23
i - Collecte des tweets	23
ii - Nettoyage	23
2.1.2. Phase d'apprentissage	23
2.2. Conception du dictionnaire	23
2.2.1. Choix du dictionnaire de base	24
2.2.2. Enrichissement du dictionnaire	25
2.3. Complétion des mots clés	25
3. Analyse d'un Tweet	26
3.1. Détection de la subjectivité	26
3.1.1. Lexique de subjectivité MPQA	26
3.1.2. Analyse de subjectivité	27
3.2. Pré-traitement	27
3.2.1. Nettoyage	27
3.2.2. Lemmatisation	27
3.2.3. Tokenisation	28
3.3. Création des trigrammes	28
3.4. Détermination du sentiment	29
3.4.1. Calcul du score	29
3.4.2. Classement de la polarité	30

CHAPITRE 4 : RÉALISATION ET MISE EN OEUVRE

PARTIE 1 : OUTILS UTILISÉS	32
1.2. Outils et langage de programmation	32
1.2.1. Python	32

1.2.1.1. Gensim	32
1.2.1.2. NLTK	32
1.2.1.3. Tweepy	32
1.2.2. PyCharm	32
1.2.3. PyQt	33
1.2.4. L'API Twitter	33
1.2.5. Twilly - outil construit -	33
PARTIE 2 : RÉALISATION DE L'APPROCHE	34
2.1. Conception du modèle Word2vec	34
2.1.1 Préparation du corpus	34
2.1.2 Phase d'apprentissage	35
2.2. Conception du dictionnaire	35
2.3. Complétion des mots clés	36
2.4. Liste des stopwords	37
2.5. Liste des mots de négation	37
2.6. Quelques exemples de code source	38
PARTIE 3 : FONCTIONNALITÉS ADDITIONNELLES	40
3.1. Examen de l'historique d'un utilisateur	40
3.2. Collecte de Tweets en temps réel	40
3.3. Interface graphique	41
 CHAPITRE 5 : EVALUATION ET DISCUSSION	
1. PARTIE 1 - DÉTECTION DE RADICALITÉ	47
2. PARTIE 2 - CATÉGORISATION DES DEGRÉS DE RADICALITÉ	48
3. PARTIE 3 - TEST SUR UN GRAND NOMBRE DE TWEETS	49
4. DISCUSSION DES RÉSULTATS	50
5. EXEMPLES DE SORTIE	51
6. SYNTHÈSE	52
7. PERSPECTIVES	52
 BIBLIOGRAPHIE	53

INTRODUCTION GÉNÉRALE

Aujourd'hui, les réseaux sociaux connaissent un succès prodigieux auprès des utilisateurs et ne cessent de progresser, ce qui a donné naissance au web interactif (web 2.0) qui désigne une évolution de l'utilisation du web dans lequel l'internaute avait juste la possibilité de consulter et d'exploiter les ressources déjà publiées, vers une utilisation participative basée notamment sur le partage de l'information, où il est lui-même capable de créer du contenu (à travers des blogs par exemple), partager et échanger des informations [34].

En 2020, il y a environ 4,5 milliards d'internautes dans le monde, dont 3,8 milliards d'utilisateurs de réseaux sociaux, tels que Facebook et Twitter, qui donnent aux utilisateurs la possibilité de communiquer, d'aborder différents sujets et d'exprimer leurs opinions spontanément [2]. Des sites se sont spécialisés dans le recueil de ces opinions dans différents domaines (*senscritique*¹...) et les internautes ont pris l'habitude de consulter les avis et les notes déposés par les autres dès qu'ils doivent prendre une décision d'achat pour un produit par exemple, ou encore pour une réservation d'hôtel [35].

Ce déferlement a touché aussi le monde professionnel où les entreprises se sont mises à améliorer leur image et à développer des relations avec leurs communautés sur les réseaux sociaux où une grande masse de données les concernant est stockée, ce qui leur permettra, en les analysant, d'avoir une idée sur l'état actuel du marché et sur les entreprises concurrentielles, pour ainsi acquérir une nouvelle stratégie de marketing et enfin établir une comparaison des produits existants dans le marché pour permettre un meilleur soutien à sa clientèle [1].

Outre le domaine du marketing, il existe d'autres domaines, comme le domaine politique ou la sécurité publique, que nous verrons par la suite, dans lesquels cette énorme masse de données présente sur les réseaux sociaux peut servir.

Néanmoins, traiter manuellement ces informations s'avère être une tâche très difficile, voire impossible vu le nombre énorme de données existant. Ce qui a permis l'apparition du domaine du traitement automatique et l'émergence des outils de data mining, comme l'analyse des sentiments.

L'analyse des sentiments propose différentes méthodes manuelles et automatiques qui permettent de prédire le sentiment de l'opinion que dégage un message avec sa polarité (positif, négatif ou neutre).

Par ailleurs, les réseaux sociaux sont considérés comme une mine d'or pour plusieurs domaines, dont ceux qui peuvent présenter un danger ou menacer et déstabiliser la vie publique comme le terrorisme, pour lequel ces plateformes sont utilisées pour inciter aux actes terroristes, pour le recrutement des organisations extrémistes et pour plus encore. En effet, des groupes terroristes comme l'Etat islamique en Irak et en Syrie (ISIS) utilisent diverses formes de médias pour diffuser de la propagande en ligne et malheureusement ces réseaux sociaux sont l'un des principaux moyens de communication de ces groupes avec une augmentation exponentielle du nombre de leurs comptes et de leurs publications chaque jour [3].

¹ <https://www.senscritique.com/films> ce site propose de découvrir, de noter et d'écrire des critiques sur des films, séries télévisées, livres, bandes dessinées, jeux vidéo, albums et morceaux musicaux.

Dans ce mémoire, notre travail consiste à proposer une méthode de l'analyse des sentiments dans le domaine du terrorisme sur les réseaux sociaux, en utilisant entre autres la notion du prolongement lexical avec le Word2vec et les trigrammes.

Pour implémenter notre approche, nous avons organisé notre mémoire en cinq grands chapitres :

- **Chapitre 1** : Ce chapitre introductif se compose de deux grandes parties. Une partie portant sur les réseaux sociaux en général, et une deuxième partie qui présente des définitions de base nécessaire à notre mémoire.
- **Chapitre 2** : Notre deuxième chapitre présente notre sujet qu'est l'analyse des sentiments, le Word Embedding, ainsi qu'une étude des travaux existants.
- **Chapitre 3** : Le troisième chapitre concerne la conception de l'approche proposée où nous parlons de toutes les étapes par lesquelles passe notre analyse.
- **Chapitre 4** : Dans le quatrième chapitre nous présentons les outils de réalisation de l'approche et nous déroulons son implémentation.
- **Chapitre 5** : Ce chapitre est dédié à l'évaluation de notre approche par rapport aux méthodes qui existe déjà, nous discutons les résultats obtenus et nous finissons par des perspectives.
- Enfin, nous terminons par une conclusion générale.

CHAPITRE 1

GÉNÉRALITÉS ET DÉFINITIONS

PARTIE 1 : GÉNÉRALITÉS SUR LES RESEAUX SOCIAUX

Les réseaux sociaux ont changé la façon de vivre du monde au 21ème siècle, ils sont désormais devenus un élément essentiel de la vie d'un individu.

L'ère numérique et l'apparition des réseaux sociaux ont permis d'accélérer la transmission de l'information et réduire le temps de réponse. Grâce à la communication en ligne, les gens du monde entier ont commencé à mieux comprendre les cultures et les traditions des autres. Cela a aussi permis à des personnes aux vues similaires de commencer à travailler ensemble pour atteindre un objectif commun [5].

Actuellement, les réseaux sociaux sont largement utilisés et ce pour divers usages. Des réseaux sociaux tels que Facebook, LinkedIn et Twitter sont couramment utilisés, partout dans le monde, pour établir des relations sociales avec d'autres personnes et partager des situations personnelles. Il existe de ce fait plusieurs plateformes de réseaux sociaux, et parmi ces plateformes, il faut distinguer deux catégories : ceux à usage exclusivement professionnel, orienté sur la mise en valeur et les échanges professionnels de ses membres, comme LinkedIn ou Viadeo. Et il y a ceux à usage privé, qui sont devenus grand-public tels que Twitter, Facebook ou Instagram [6].

1.1. Web 2.0

Le Web 2.0, également appelé le Web interactif ou Web participatif, désigne l'ensemble des techniques, des fonctionnalités et des usages qui ont suivi la forme originelle du web, *www* ou *World Wide Web*, caractérisée par plus de simplicité et d'interactivité. Elle concerne en particulier les interfaces et les échanges permettant aux internautes ayant peu de connaissances techniques de s'approprier de nouvelles fonctionnalités du web [37].

Grâce au Web 2.0, les internautes peuvent contribuer à l'échange d'informations et interagir de façon simple, à la fois au niveau du contenu et de la structure des pages, et aussi, entre eux, créant notamment le Web social. L'internaute devient, grâce aux outils mis à sa disposition, une personne active sur la toile [37].

On peut donc dire que les réseaux sociaux sont des sites qui parviennent de cette vague du Web 2.0 (YouTube, Facebook, Dailymotion...) [7].

1.2. Réseaux sociaux

Dans le cadre du Web, le terme de "réseaux sociaux" désigne généralement l'ensemble des sites internet permettant de se constituer un réseau d'amis ou de connaissances professionnelles et fournissant à leurs membres des outils et interfaces d'interactions, de présentation et de communication [8].

Il existe aujourd'hui énormément de différents réseaux sociaux, nous allons aborder ci-dessous quelques-uns des plus connus et des plus utilisés.

1.2.1. Twitter

Twitter est un réseau social dit de « microblogging » qui permet de communiquer sous la forme de messages courts ne dépassant pas 280 caractères appelés « Tweets ». Ils peuvent contenir des URL sous forme raccourcie, des images, des émoticônes, des gifs animés et des vidéos.

Le service a été créé en 2006 par Jack Dorsey, Evan Williams, Biz Stone et Noah Glass. Le premier Tweet a été envoyé par Jack Dorsey le 21 mars 2006. Initialement pensé comme un moyen de partager facilement des moments de vie, Twitter s'est rapidement imposé comme un outil permettant de suivre l'actualité en temps réel.

Célébrités, marques, politiciens, mais aussi les médias traditionnels se sont appropriés ce réseau social qui leur permet d'établir une communication plus directe avec le public. Lorsqu'un événement important se produit dans le monde, c'est généralement sur Twitter qu'il est relayé en premier [10].

1.2.2. Facebook

Facebook est un réseau social conçu pour « connecter » les gens les uns aux autres par le biais de publications personnelles, de partage d'informations, d'actualités ou de contenu audiovisuel.

L'idée de ce réseau social consiste à établir un contact avec des personnes proches de l'utilisateur, telles que des amis et de la famille, des collègues, ou des personnes qui ont des intérêts similaires aux siens. Il compte actuellement environ 2,4 milliards d'utilisateurs répartis dans le monde, et connaît une croissance exponentielle quasiment depuis ses débuts [11].

1.2.3. Instagram

Instagram est un réseau social gratuit construit autour du partage de photos et de vidéos. Il a été lancé en octobre 2010 sur iOS pour la première fois et est devenu disponible sur Android en avril 2012. Facebook a acheté le service en avril 2012 et en est propriétaire depuis.

Comme la plupart des réseaux sociaux, Instagram permet à l'utilisateur de suivre les autres utilisateurs qui l'intéressent. Ceci crée un flux sur la page d'accueil montrant les posts récents de tous ceux qu'il suit. L'utilisateur peut aimer ces posts et les commenter, il peut également échanger via des messages privés avec les autres membres [36].

1.2.4. LinkedIn

LinkedIn est un réseau social créé en 2002. Il fait partie des réseaux sociaux les plus importants au monde et se différencie de ceux destinés à un usage personnel, comme ceux vu précédemment.

L'idée consiste à connecter les professionnels du monde entier pour les aider dans l'avancement de leurs carrières. On y retrouve plus de 150 millions d'utilisateurs actifs, venant de 200 pays différents.

LinkedIn permet à chaque utilisateur d'avoir sa page professionnelle personnalisée, comme un CV, où il partage ses études, compétences, formations et expériences professionnelles dans le but de se créer un réseau de connaissances et faire avancer sa carrière [9].

PARTIE 2 : DÉFINITIONS

2.1. Définition d'une émotion

L'émotion est la réaction de l'individu vis-à-vis un événement qui lui arrive [12].

2.2. Définition d'un sentiment

Le sentiment est l'impression laissée par cette émotion après cet événement. C'est aussi défini comme étant ce que l'on perçoit d'une situation et les pensées que nous y associons [13].

2.3. Définition d'une opinion

Avis personnel, manière de penser sur un sujet ou un ensemble de sujets [14]. C'est aussi l'ensemble de jugements que l'on porte sur un objet [15].

2.4. Définition de la polarité d'une opinion

On peut dire d'un avis ou d'une opinion qu'elle est positive, négative ou neutre. Mais ce qu'on veut dire par polarité, c'est la variation des degrés de positivité ou de négativité [16] (voir Figure 1).

Donc la polarité peut être positive, négative et neutre ou bien elle peut être définie par le degré de positivité (ou de négativité), par exemple on peut fixer des valeurs entre « 1 – 5 », dont 1 est la polarité la plus négative et 5 la polarité la plus positive, ceci est aussi appelé orientation sémantique.

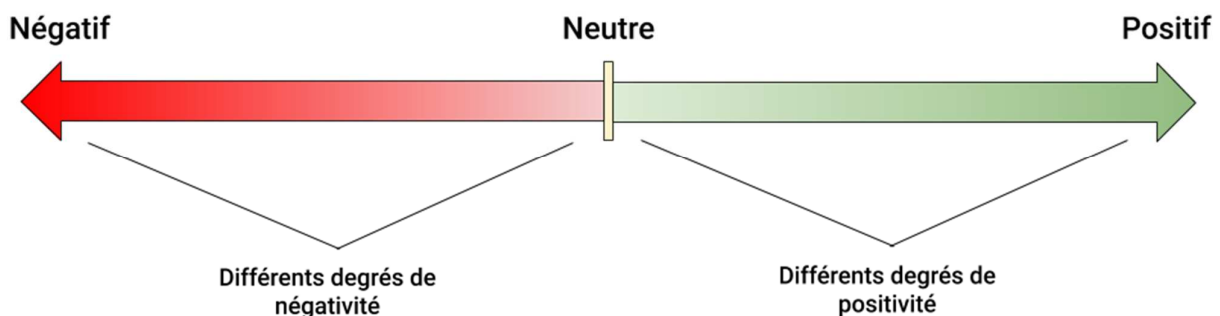


Figure 1 – Polarité.

2.5. Définition d'une expression subjective

Les expressions subjectives sont des mots ou des phrases utilisés pour exprimer des états personnels, c'est-à-dire des états mentaux et émotionnels tels que des spéculations, opinions, sentiments et croyances [43].

2.6. Définition de l'analyse des sentiments

L'analyse des sentiments est un domaine de recherche qui permet d'analyser des données et de détecter l'opinion ou l'avis, avec la polarité, qu'elles dégagent envers une personne, un produit ou un événement [18]. Elle se réfère donc, à l'extraction automatique du texte qui aide à produire des résultats prédictifs. Bing Liu a présenté une définition de l'analyse des sentiments comme étant « ce domaine de l'étude qui analyse les opinions, les sentiments, les attitudes et les émotions des gens envers des entités telles que des produits, des services, des organisations, des problèmes, des événements et des sujets ». Elle représente donc un grand espace de recherche [19].

L'analyse de sentiment est le terme le plus utilisés pour désigner ce domaine d'étude mais plusieurs appellations lui sont associé dont opinion mining (fouille de données) ou analyse d'émotion.

Ce domaine de la fouille de données a connu un succès grâce aux masses de données présentes sur les réseaux sociaux qui permettent aujourd'hui d'analyser des big data pour construire des modèles de prédiction de polarité de plus en plus précis.

2.7. Définition de la radicalité

La radicalité ou la radicalisation est définie comme étant l'adoption progressive et évolutive d'une pensée rigide, vérité absolue et non négociable, usant de la violence pour la faire entendre, le plus souvent au sein de structures clandestines.

Nous utiliserons ce mot dans la suite du mémoire pour exprimer tout ce qui encourage ces organisations extrémistes et qui incite à les rejoindre et à commettre des actes terroristes.

CHAPITRE 2

ÉTAT DE L'ART

PARTIE 1 : L'ANALYSE DES SENTIMENTS

L'analyse des sentiments est un domaine devenu important en informatique. Apparu au début des années 2000, ce domaine se développe depuis pour sonder de mieux en mieux l'avis public sur un sujet spécifique [17].

1.1. Les niveaux de l'analyse des sentiments

Principalement, on distingue trois niveaux de l'analyse des sentiments [18] :

1.1.1. Niveau document

Ce niveau permet d'analyser un document comme une seule entité et d'en tirer à la fin le sentiment exprimé. Si on suppose qu'on a plusieurs documents alors ils seront traités séparément, comme par exemple, plusieurs documents où chacun traite un produit différent. Cette méthode reste applicable jusqu'au moment où un même document traitera différents sujets même s'ils font partis d'une même catégorie (par exemple : un même document qui aborde différentes caractéristiques d'un produit).

1.1.2. Niveau phrase

Contrairement au traitement du niveau document, le niveau phrase consiste en la détermination de l'opinion positive, négative ou neutre de chaque phrase. Ce niveau joue un rôle important dans l'analyse des sentiments dont l'intérêt est de distinguer les phrases qui expriment une opinion subjective ou des faits réels, mais cela n'est pas toujours faisable car, prenons par exemple la phrase citée dans [18] : « on a acheté la voiture le mois dernier et l'essui glace est déjà tombé » cette phrase est objective mais exprime une opinion négative.

1.1.3. Niveau entité et aspect

Nous remarquons que les deux niveaux précédents procèdent de la même manière, c'est-à-dire qu'ils s'intéressent directement à la construction de la donnée (phrase, texte, document ...) or cela n'aide pas et ne détermine pas exactement ce que pense ces personnes. Le niveau aspect s'intéresse directement à l'opinion elle-même en se basant sur une règle : une opinion a un sentiment (positive ou négative) et une cible. Pour mettre en valeur l'utilité de connaître la cible de l'opinion, on étudie la phrase suivante : « Même si le service n'est pas super, j'aime toujours ce restaurant » cette phrase a une partie positive certes mais on ne peut pas dire qu'elle est entièrement positive. « J'adore le restaurant » exprime un sentiment positif par rapport au restaurant mais « le service n'est pas super » est un avis négatif concernant le service. Dans plusieurs cas, la cible est identifiée par plusieurs entités (aspects), comme dans le cas « la qualité d'appel de l'iPhone est bonne mais sa batterie ne dure pas longtemps » on distingue ici deux entités d'aspects : la qualité d'appel et la batterie.

1.2. Le processus de l'analyse des sentiments

Le processus de l'analyse des sentiments passe généralement par quatre étapes qui sont [24] :

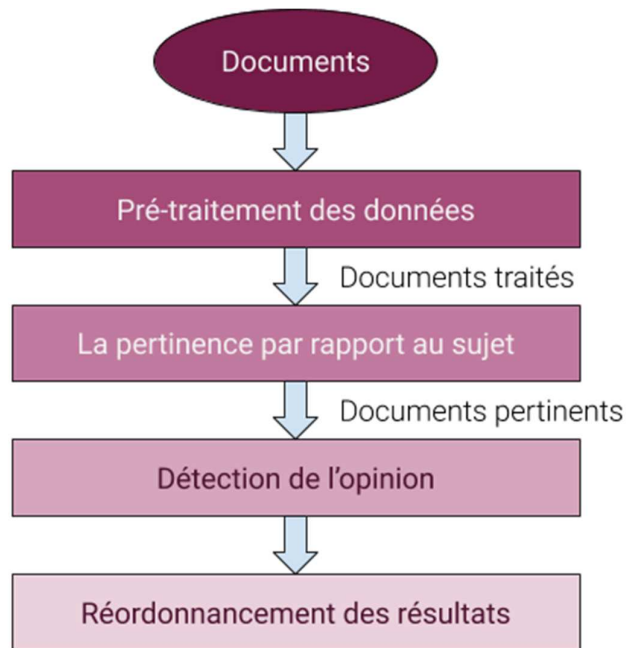


Figure 2 - Processus de fouille d'opinions.

1.2.1. Acquisition et prétraitement du corpus

Dans cette phase, les textes, une fois acquis, sont prétraités, puis une analyse lexicale est effectuée pour enlever les mots redondants. Ces traitements sont effectués selon des besoins et des objectifs.

1.2.2. La pertinence par rapport au sujet

Cette étape consiste à étudier la pertinence des documents par rapport au sujet étudié. Les documents sont classés et utilisés pour l'étape suivante : la détection d'opinions.

1.2.3. La détection d'opinions

Plusieurs méthodes sont utilisées pour la détection d'opinions. Leur but est d'affecter un score d'opinion à chaque document.

1.2.4. Réordonnancement des résultats

Cette étape consiste à classer les textes selon leurs scores.

1.3. Les domaines d'application de l'analyse des sentiments

Il existe plusieurs travaux qui ont traité l'analyse des sentiments pour distinguer les opinions des utilisateurs sur un sujet donné, et ce dans plusieurs domaines. On a subdivisé ces derniers en quatre grandes parties :

1.3.1. Politique

De nombreuses études dans le domaine de la politique ont été réalisées dans le but de détecter l'avis public envers des événements politiques, comme par exemple les sentiments exprimés envers les candidats en vue d'une élection, ou concernant des sujets controversés afin de passer de nouvelles lois (ex : avortements, le port d'arme, la peine de mort...).

Une de ces études [21] propose une analyse de sentiments sur des Tweets contenant la chaîne de caractères 'Obama' dans le but de jauger l'opinion publique sur le désormais ex-président des Etats-Unis Barack Obama, lorsqu'il était en poste. Pour cela, les auteurs ont étudié les méthodes d'apprentissage supervisées et semi-supervisées. Leur approche implique l'utilisation d'un classifieur basé sur les n-grammes afin d'exploiter les relations entre ces derniers, les Tweets et les utilisateurs.

1.3.2. Marketing

Avant d'acheter un produit, la majorité des clients demandent conseil sur un produit ou un service donné et sont même disposés à payer plus pour un produit dont l'opinion est plus favorable qu'un autre, ce qui peut augmenter les ventes. Grâce à l'analyse des sentiments, les entreprises peuvent connaître l'opinion des clients sur leurs produits ou leurs services, dans une perspective d'amélioration de leurs produits et d'augmentation de leurs ventes et revenus [55].

Les auteurs dans [22] ont proposé une étude permettant de classifier les Tweets relatifs aux avis des utilisateurs sur les services des compagnies aériennes américaines en trois classes : positive, négative ou neutre. Pour ce faire, ils ont pris un échantillon de Tweets, à partir du site kaggle.com, qui sont déjà classés, ils ont ensuite divisé cet échantillon en 2 parties : une partie dédiée à l'apprentissage (70%) et une partie de test (30%), pour appliquer par la suite la méthode Word2vec sur les données d'apprentissage pour apprendre le contexte et produire des mots similaires dans le but d'améliorer les résultats de prédiction. Pour cette dernière, ils ont utilisé deux classifieurs qui sont SVM et la régression logistique.

1.3.3. Cyberharcèlement

Le cyberharcèlement est malheureusement bien présent sur les réseaux sociaux, et plusieurs études ont été menées dans le but de le détecter et de l'atténuer. L'étude [23] porte sur la détection automatique de la cyberintimidation en exploitant les caractéristiques psychologiques, y compris les personnalités, les sentiments et les émotions des utilisateurs de Twitter.

Les personnalités des utilisateurs ont été déterminées à l'aide des modèles Big Five¹ et Dark Triad², tandis que les classificateurs d'apprentissage automatique ont été utilisés pour classer les tweets dans l'une des quatre catégories suivantes : intimidateur, agresseur, spammeur et normal. Le jeu de

¹ Modèle descriptif de la personnalité utilisé en psychologie.

² Test psychologique qui détecte les traits de narcissisme, machiavélisme et la psychopathie.

données Twitter contenait 5453 tweets rassemblés à l'aide de l'hashtag #Gamergate et annotés manuellement par des experts.

1.3.4. Sécurité publique

L'analyse des sentiments permet d'aider les autorités à détecter et de prédire d'éventuels actes terroristes, elle aide également à empêcher la propagande et à nuire au réseau de recrutement des groupes. Une étude menée par [4] propose une méthode pour détecter du contenu radical sur Twitter.

Pour ce faire, ils ont collecté des Tweets en langue arabe relatifs aux activités terroristes, puis ils les ont classifiés en deux classes : incitant au terrorisme, ou non incitant au terrorisme. Ils ont ensuite utilisé les méthodes Word2vec classique, et Word2vec par poids moyen pour représenter les Tweets, puis, deux algorithmes d'apprentissage automatique - SVM et Random Forest - ont été utilisés pour prédire les sentiments puis comparer les résultats des deux méthodes.

1.4. Les problèmes de l'analyse des sentiments

Analyser un sentiment se fait généralement par polarité positive, négative ou neutre. Cette classification n'est pas toujours une tâche facile et s'avère présenter plusieurs difficultés relatives au langage naturel que nous avons résumé en trois problèmes principaux :

1.4.1. Problèmes liés au contexte

Prédire le sentiment qu'une phrase dégage en se basant sur un certain groupe de mots représente lui-même une difficulté car le sens des mots peut changer d'un domaine à l'autre.

La phrase suivante illustre qu'un mot ou une phrase peut exprimer des opinions différentes selon le contexte. - « *Go read the book* » Si la phrase figure dans la critique d'un livre, elle exprime un sentiment positif, étant donné qu'elle recommande sa lecture. En revanche, si la phrase figure dans la critique d'un film, elle exprime un sentiment négatif à l'égard du film, car elle incite la lecture du livre parce que l'auteur de la phrase suggère que le livre est meilleur que son adaptation [26].

Le contexte ne change pas seulement en dépendance du sens des mots, on trouve parfois deux avis qui s'opposent dans une même phrase, et l'opinion finale dépend du contexte comme dans la phrase « j'aime ce film mais ma sœur le trouve mauvais », dans ce cas on a deux avis différents, si on s'intéresse à l'opinion de l'auteur le sentiment déduit est positif « j'aime ce film », sinon si on s'intéresse à l'opinion de la sœur le sentiment est négatif [24].

1.4.2. Problèmes liés à la structure du texte

Les publications et les commentaires sur les réseaux sociaux ont rarement une structure de texte correcte. Ces commentaires libres contiennent généralement plusieurs phrases composées de nombreux éléments considérés comme du « bruit »¹. Ce bruit nécessite d'être identifié et éliminé [25].

¹ C'est toute information non pertinente qui ne contient pas de sentiment d'expression et qui ne précise pas le contexte.

1.4.3. Problèmes liés aux fautes lexicales

Des fautes lexicales et grammaticales peuvent se glisser en plus du manque de la ponctuation et des abréviations (“dunno” pour “don’t know”, “y” pour “why”...) qu'utilisent les internautes entre eux, ceci est dû parfois au nombre de caractères limités (280 caractères sur Twitter par exemple) [24].

PARTIE 2 : ETAT DE L'ART

2.1. LES DIFFÉRENTES APPROCHES DE L'ANALYSE DES SENTIMENTS

On distingue principalement 3 approches pour l'analyse des sentiments - *Figure 3* - :

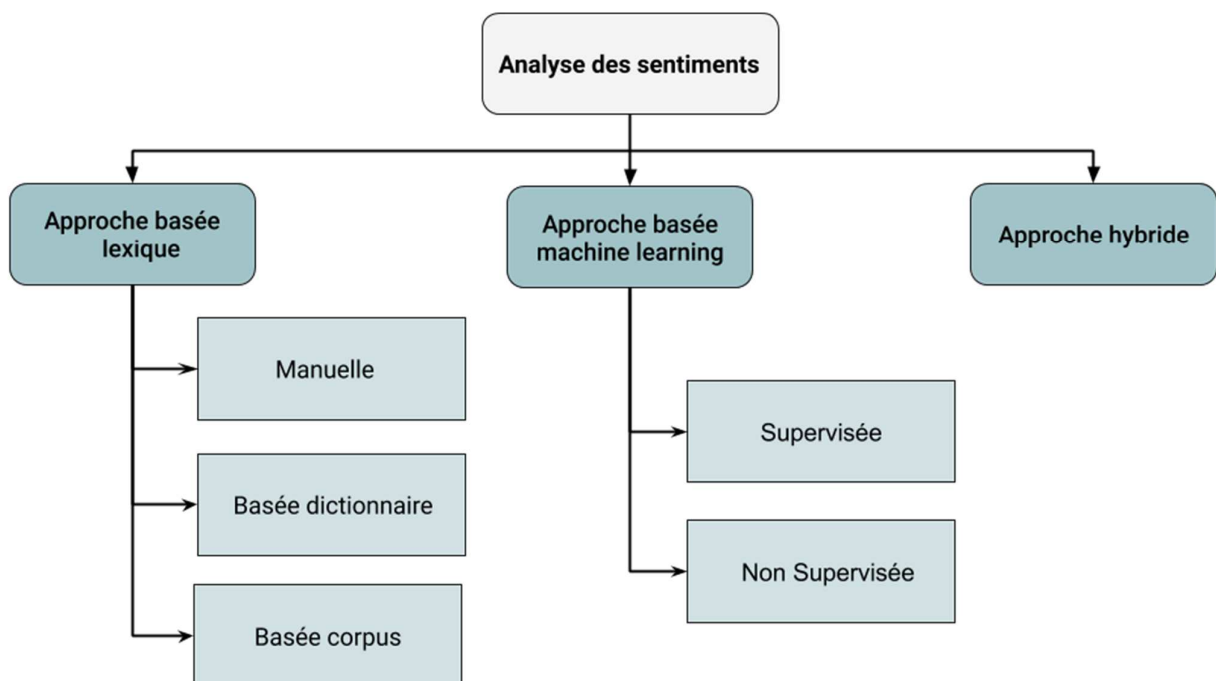


Figure 3 - Les différentes approches de l'analyse des sentiments.

2.1.1. Approche basée sur le lexique

Cette approche consiste à utiliser un dictionnaire qui contient des mots d'opinion avec leurs poids respectifs et les fait correspondre avec les données afin de déterminer leurs polarités. Elle attribue les scores de sentiment aux textes analysés décrivant s'ils sont positifs, négatif ou neutre [27].

Il existe trois méthodes pour construire le lexique [42] :

- Méthode manuelle : cette approche consiste en la construction manuelle des termes et leurs poids, cette approche est difficile car elle nécessite beaucoup de ressources humaines et prend du temps.
- Méthode basée sur un dictionnaire : cette approche consiste à utiliser un dictionnaire déjà construit, et de se baser dessus afin d'effectuer l'analyse. Il est donc nécessaire que le dictionnaire soit adapté au domaine étudié.
- Méthode basée sur un corpus : cette approche est basée sur l'exploration des mots dans un corpus. Le principal avantage de cette approche est qu'elle permet de découvrir des termes sentimentaux dans un contexte spécifique.

2.1.2. Approche basée apprentissage automatique

Utilise des algorithmes de classification pour classer les textes en différentes classes, et pour cela, il existe deux techniques :

2.1.2.1. Supervisée

Implique la présence de deux ensembles de données, un ensemble étiqueté présenté à la machine pour l'apprentissage appelé « ensemble de données d'entraînement », et un second ensemble dédié aux tests. Elle est dite supervisée puisque la machine est entraînée par des données déjà traitées [28]. Cette technique nécessite la collecte des données d'apprentissage et de les étiqueter, puis d'entraîner un classifieur sur ces données, avant de le tester sur l'ensemble de test, afin d'évaluer sa précision. Il existe plusieurs classifieurs d'apprentissage automatique comme Naïve Bayes (NB), l'entropie maximale (ME), et les machines à vecteurs de support (SVM) qui ont connu un grand succès dans l'analyse des sentiments [27].

2.1.2.2. Non supervisée

Ne recommande qu'un seul gros ensemble de données et donc exige que le système lui-même réorganise ces données d'une manière à ce que les informations les plus similaires soit dans un même groupe [28].

2.1.3. Approche hybride

Cette approche combine les points forts des deux approches précédentes. Elle prend en compte tout le traitement linguistique des approches lexicales avant de lancer le processus d'apprentissage comme dans les approches basées Machine Learning [29].

2.2. LE WORD EMBEDDING DANS L'ANALYSE DES SENTIMENTS

Le Word Embedding, ou prolongement lexical, est une méthode d'apprentissage automatique qui concerne la représentation des mots. Il désigne le processus de transformation des mots en format numérique qu'on appelle des vecteurs. Chaque mot est représenté par un vecteur à valeur réelle [38].

La particularité de cette méthode est que les mots qui apparaissent dans des contextes similaires ont des vecteurs à valeurs très proche. Par exemple, on pourrait s'attendre à ce que les mots « King » et « Queen » soient représentés par des vecteurs proche dans l'espace vectoriel¹ où ils sont définis [39].

Il existe plusieurs techniques qui permettent de produire le Word Embedding dont le Word2Vec.

2.2.1. Le Word2Vec

Le Word2vec est parmi les algorithmes les plus connus du Word Embedding. Il a été développé en 2013 par une équipe de recherche de Google sous la direction de Tomas Mikolov. Il prend en entrée un corpus de texte non étiqueté et cherche à apprendre les représentations vectorielles des mots qui compose le corpus pour produire en sortie les vecteurs des mots de manière à ce que les mots qui ont des contextes similaires soient représentés par des vecteurs numériques proches [40].

Le Word2Vec possède deux architectures, CBOW et Skip-Gram, l'utilisateur est libre de choisir l'architecture qui répond le mieux à ses besoins, le tableau suivant présente la particularité de chacun d'eux [41].

CBOW	Skip-Gram
<ul style="list-style-type: none">- Plus rapide que Skip-gram.- Donne une meilleure importance pour les mots les plus fréquents.	<ul style="list-style-type: none">- Nécessite une petite quantité de données d'entraînement.- Prend en compte les mots peu répétés.

Tableau 1 - Comparaison entre CBOW et Skip-Gram.

2.2.1.1. CBOW

Cet algorithme reçoit en entrée le contexte d'un mot, c'est à dire les termes qui l'entourent dans une phrase, et essaye de prédire le mot en question.

¹ Un **espace vectoriel** est un ensemble de vecteurs, que l'on peut additionner entre eux, ou même multiplier par un réel.

La Figure 4 illustre l'architecture CBOW, appliquée au pangramme¹ anglais « *the quick brown fox jumps over the lazy dog* ».

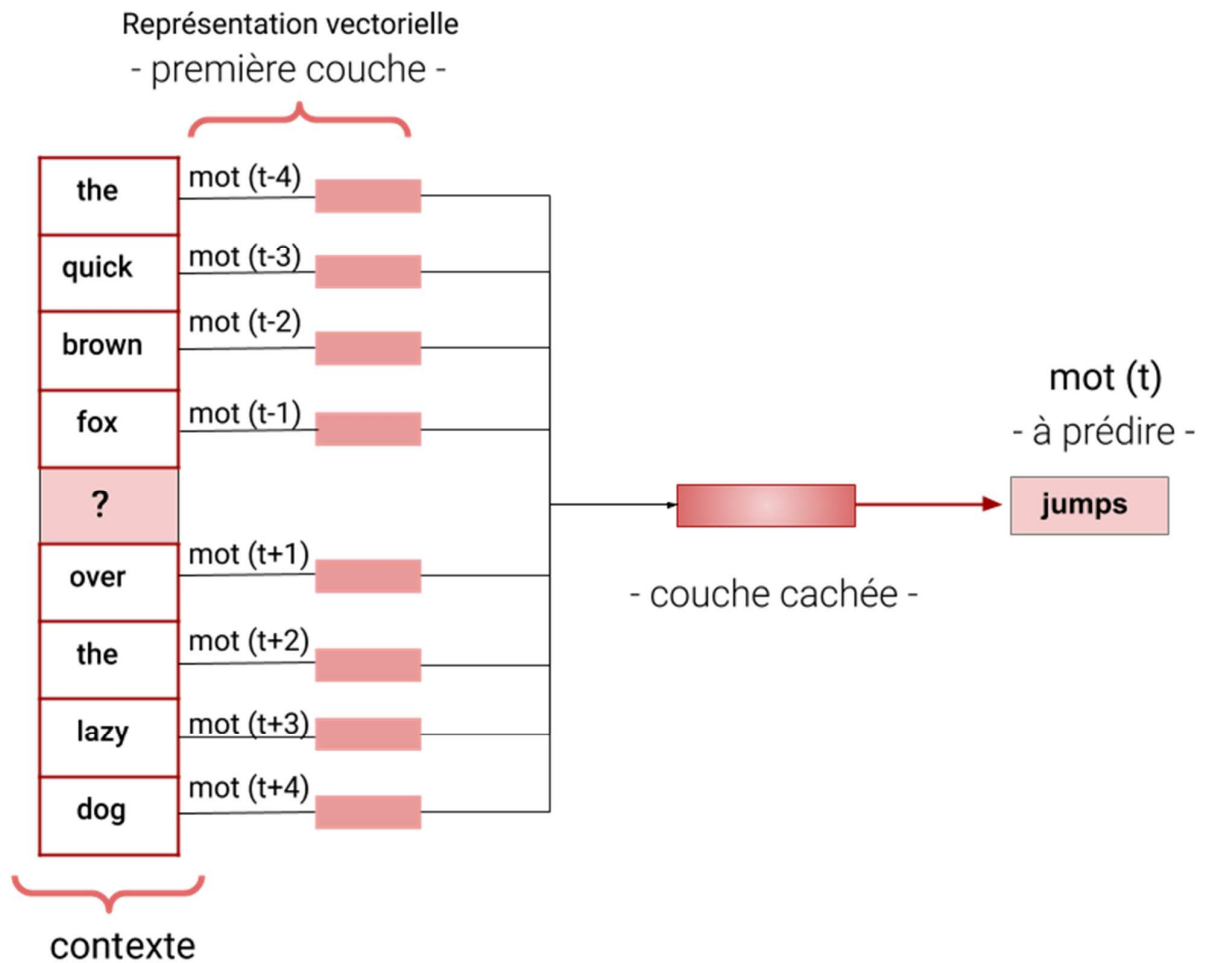


Figure 4 - Architecture CBOW.

Comme nous pouvons le constater, la première couche projette chaque mot du contexte vers sa représentation vectorielle. Puis la couche cachée analyse ces représentations vectorielles afin de tenter de prédire le mot central.

2.2.1.2. Skip-Gram

Cet algorithme fait exactement le contraire : il prend en entrée un mot et essaye de prédire son contexte.

La Figure 5 illustre le même exemple utilisé dans la Figure 4 en appliquant l'algorithme Skip-Gram.

¹ Phrase ayant la caractéristique de contenir toutes les lettres de l'alphabet.

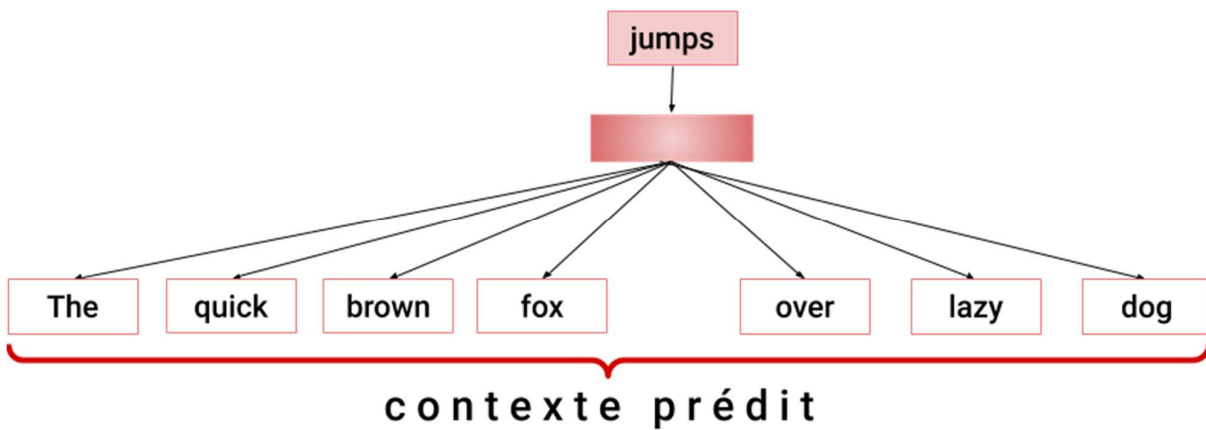


Figure 5 - Architecture Skip-Gram.

Dans les deux cas, l'entraînement du modèle se fait en parcourant le texte fourni.

Word2Vec possède différents paramètres que l'utilisateur doit fixer, les plus importants sont :

- La dimensionnalité de l'espace vectoriel à construire (*size*) : c'est à dire le nombre de descripteurs numériques utilisés pour décrire les mots (entre 100 et 1000 en général).
- La taille du contexte d'un mot (*window*) : c'est à dire le nombre de termes entourant le mot en question (les auteurs suggèrent d'utiliser des contextes de taille 10 avec l'architecture Skip-Gram et 5 avec l'architecture CBOW).

Étant donné que le Word2Vec n'est composé que de deux couches, cet algorithme est donc très rapide à entraîner et à exécuter, ce qui se révèle être un avantage important par rapport à d'autres méthodes du Word Embedding. C'est pour cela qu'il existe plusieurs travaux de l'analyse des sentiments l'utilisant, notamment pour enrichir le lexique utilisé ou pour le construire à partir de zéro dans le domaine étudié [4][22].

2.5. LES TRAVAUX EXISTANTS

Dans cette partie, nous allons présenter quelques travaux basés sur l'analyse des sentiments sur réseaux sociaux :

2.5.1. Conde-Céspedes et al. [30]

Les auteurs ont présenté une méthode basée Machine Learning dans le but de détecter les comptes soupçonnés de partager de la propagande terroriste.

Pour ce faire, ils ont recueilli les tweets publiés avant et peu après les attentats terroristes de Paris, en novembre 2015. Pour construire l'ensemble des tweets à étudier, ils ont commencé par identifier les mots et les expressions courantes utilisés par les PRO-ISIS comme #toParisWithRevenge, #jesuiskouachi...

Les Tweets collectés ont été labellisés manuellement dans les 2 classes suivantes : Tweet Pro-ISIS ou Tweet neutre. Les Tweets qui étaient en français ou en arabe ont ensuite été traduits vers l'anglais à l'aide de Google Traduction ainsi que d'autres outils supplémentaires pour minimiser le taux d'erreur.

Enfin, ils ont utilisé ces données entraîner le classifieur SVM. Les résultats étaient concluants puisque l'exactitude moyenne des résultats était de 87,9%.

2.5.2. Asif et Ishtiaq [31]

Le but de ce travail est de détecter le contenu extrémiste dans les messages et dans les commentaires de diverses pages publiques, sur les réseaux sociaux au Pakistan.

Pour cela, les auteurs ont utilisé Facebook, car parmi les 35 millions d'utilisateurs de réseaux sociaux au Pakistan, 32 millions sont sur Facebook. L'outil "*Data Miner Scraper*"¹ a été utilisé pour collecter les données.

Comme la majorité de la population du Pakistan écrit en anglais, en ourdou ou en roumain-ourdou pour s'exprimer sur Facebook et en raison de l'indisponibilité d'un dictionnaire de lexique extrême multilingue, les auteurs ont créé un dictionnaire à partir de zéro comprenant des termes qui expriment différents niveaux de sentiments extrêmes.

Après la phase de prétraitement, ils ont utilisé la méthode TF-IDF pour attribuer un poids aux mots importants, pour une classification appropriée du contenu.

Enfin, les résultats de l'analyse des sentiments effectuée en utilisant différents algorithmes supervisés et non supervisés, sur les classifieurs Naïve Bayes Multinomial et SVM, ont révélé que les algorithmes supervisés sont beaucoup plus précis que les non supervisés.

2.5.3. Sohail et Qadoos [32]

Dans ce travail, les auteurs ont utilisé une approche d'apprentissage automatique afin de détecter les activités des groupes terroristes sur Twitter.

L'ensemble de données a été collecté, puis a été nettoyé en supprimant les mots inutiles et les symboles supplémentaires.

Pour effectuer l'analyse, les auteurs ont utilisé le lexique Sentiwordnet et la bibliothèque TextBlob.

Ensuite, ils ont entraîné les données sur différents classifieurs afin d'obtenir la précision maximale. La précision était de 83% avec le classifieur Random Forest et de 89% avec le classifieur Naïve Bayes multinomial.

¹ Data Miner est une extension de navigateur Google Chrome et Microsoft Edge qui aide à extraire des données des pages Web dans un fichier CSV ou une feuille de calcul Excel.

2.5.4. Rattrout et Ateeq [33]

Les auteurs ont proposé un modèle hybride qui évoque le problème de l'analyse des sentiments multi-domaines. Ce modèle utilise l'approche basée sur le dictionnaire et la logique floue.

Le but de cette étude est de créer un prototype qui calcule les polarités des phrases collectées et les classer en sept catégories : 'très positive', 'positive', 'bonne', 'neutre', 'pas bonne', 'négative' et 'très négative'.

Pour atteindre cet objectif, ces auteurs ont utilisé le Framework "ASP.Net"¹ et le paquet "LinqToTwitter"² pour se connecter via l'API Twitter. Pour collectés les données de Twitter et Facebook, un ensemble de comptes qui publient en langue arabe a été sélectionné. La liste comprenait des agences de presse et des personnalités célèbres. Les données sont passées par la phase de prétraitement avant de commencer le processus d'analyse qui a été divisé en deux phases principales.

Pour calculer la polarité d'une phrase, les auteurs se sont basés sur la polarité des mots et des émoticônes qui la composent. La première phase consiste à utiliser le dictionnaire SentiWordNet pour obtenir la polarité de chaque mot de la phrase. Alors que ce dictionnaire est dédié aux mots anglais, Google Translate a été utilisé pour effectuer la traduction vers l'arabe. Ensuite, les émoticônes ont été collectés manuellement du site "emojipedia.org", la polarité de chaque émoticône a été calculée à l'aide de SentiWordNet en agrégeant les polarités pour les termes de description des émoticônes.

La seconde phase consiste à utiliser la logique floue pour obtenir la classification des phrases finales en utilisant trois facteurs qui sont : la polarité de la phrase à l'aide du dictionnaire, le ratio de likes de phrases³ et l'orientation du compte⁴.

Pour l'évaluation, les auteurs ont collecté deux échantillons de données à partir de Twitter, ensuite chacun a été testé deux fois. Les résultats obtenus ont révélé que le système apprend et modifie en permanence la classification si et seulement s'il y a de nouvelles données d'entrée et ce pour mettre à jour le taux de likes et le ratio d'orientation du compte.

2.5.5. Jamilah Rabeh Alharbi et al. [42]

Dans cette étude, les chercheurs ont choisi comme échantillon d'étude des avis de consommateurs collectés sur Twitter concernant les fournisseurs des services Cloud.

Le but de cette étude est de mesurer la satisfaction des utilisateurs concernant les services Cloud fournis par Google, Microsoft et Amazon. L'analyse des Tweets des utilisateurs présentera une vue générale du niveau de satisfaction et des opinions des clients, ce qui donnera aux fournisseurs des informations sur les améliorations possibles.

¹ Est un Framework permettant de générer à la demande des pages web.

² Permet de faire des requêtes pour Twitter.

³ Une valeur variable qui mesure la popularité d'une phrase spécifique en utilisant le nombre de ses likes dans une comparaison avec le nombre maximum de likes obtenu par toutes les phrases pour le même compte.

⁴ Mesure variable qui indique l'orientation générale d'une page spécifique d'un point de vue sentimental.

L'approche proposée est une approche hybride utilisant les dictionnaires SentiWordNet et SentiStrength ainsi que la logique floue pour classer avec précision les Tweets en fonction de leur contenu en cinq catégories : 'très positif', 'positif', 'neutre', 'négatif' et 'très négatif'.

Après le prétraitement des Tweets collectés grâce à l'API de Twitter à partir des comptes de Google, Microsoft et Amazon ; un score est attribué à chaque Tweet en utilisant les deux dictionnaires et en suivant une liste de critères.

Le contrôleur "*Fuzzy Logic Takagi-Sugeno-Kang (TSK)*" a été utilisé pour finaliser la classification. Dans cette étape, la sortie doit être un terme linguistique au lieu d'une valeur réelle.

Les résultats ont montré que l'approche proposée atteignait une précision de 83%, un rappel de 89% et un F-mesure de 83%, sur un échantillon annoté manuellement.

2.5.6. Yun Wang et al. [54]

Dans cette étude, les auteurs visent à développer un système d'analyse des sentiments basé sur la logique floue dans le but d'analyser les commentaires des étudiants afin de quantifier leur satisfaction par rapport aux cours, le corps professoral... pour des futures améliorations.

La technique proposée calcule le score de sentiment des commentaires des étudiants collectés à partir de Twitter, puis détecte les Tweets subjectifs en se basant sur plusieurs lexiques de subjectivité afin de ne garder que les Tweets qui comportent un point de vue et ce, après la phase de nettoyage. Puis, les auteurs ont appliqué un module de logique floue pour analyser et quantifier la satisfaction des étudiants.

Les résultats expérimentaux révèlent que le travail proposé a surpassé les études de base ainsi que les classificateurs d'apprentissage automatique.

2.5.7. D'autres travaux dans le domaine du terrorisme

En raison du grand chevauchement entre le monde virtuel et la vie réelle, les données des réseaux sociaux ont été utilisées pour prédire par exemple les résultats des élections, détecter la propagation des épidémies de grippe ou comme source pour trouver des témoins oculaires lors d'incidents criminels et bien plus encore [59]. Ceci a également fortement affecté le mode de fonctionnement des cybercriminels et des groupes extrémistes, des études montrent qu'Internet et les médias sociaux ont joué un rôle important dans l'augmentation de l'extrémisme [60] par exemple les groupes radicaux comme ISIS, Al-Qaïda et Daech ont énormément profité de ces plateformes car ils leur offrent des opportunités de diffuser leurs propagandes et de faciliter le recrutement de potentiels nouveaux membres [61].

Pour limiter l'activité des cyber-terroristes, plusieurs organisations privées et gouvernementales contrôlent le contenu en ligne et utilisent des technologies de mégadonnées pour minimiser les dommages et contrer la propagation de ces informations.

En 2010 par exemple, le Royaume-Uni a lancé une unité de référence contre le terrorisme sur Internet dans le but de supprimer le contenu illégal et aider la police à enquêter sur les activités terroristes et les activités de radicalisation en ligne. L'unité rapporte que parmi les liens les plus fréquemment mentionnés provenaient de plusieurs réseaux sociaux, tels que Facebook et Twitter. De même, plusieurs de ces plateformes en ligne travaillent constamment à la détection et à la suppression des utilisateurs faisant la promotion de contenus extrémistes [60].

En examinant plus la littérature existante, nous constatons qu'un certain nombre d'études dans ce domaine analysent les propriétés psychologiques qui peuvent être déduites des utilisateurs, ce processus est appelé psycholinguistique, où l'on examine comment l'utilisation de la langue peut être indicative de différents états psychologiques (l'introversion, extraversion...) [53].

SYNTHÈSE

Dans ce chapitre, nous avons abordé le domaine de l'analyse des sentiments en étudiant les nombreux aspects qui la caractérisent, nous avons par la suite détailler les différentes méthodes utilisées dans ce domaine. Nous avons également abordé la technique du Word Embedding, et plus précisément le Word2vec. Nous avons terminé par examiner plusieurs travaux existants.

Conscients du danger de l'extrémisme violent, de la radicalisation et de la façon dont cela devient un défi majeur pour les sociétés du monde entier, nous avons décidé d'étudier le domaine du terrorisme et de radicalité en ligne.

S'appuyant sur les résultats et les efforts des études précédentes, notre mémoire vise à réaliser une analyse textuelle basée lexicale, en utilisant les trigrammes ainsi que le Word2Vec, dans le domaine du terrorisme, sur des données recueillies à partir de Twitter.

CHAPITRE 3

CONCEPTION DE L'APPROCHE

Introduction

L'approche que nous proposons se base sur la méthode lexicale basée dictionnaire. Nous avons donc choisi un dictionnaire de termes pondérés et nous l'avons enrichi en utilisant un modèle Word2vec, qui est une technique du Word Embedding. Nous avons également utilisé ce même modèle pour enrichir une liste de mots-clés reliée au terrorisme qui nous aideront dans la construction des trigrammes, et tout cela dans le but d'analyser les données provenant du réseau social Twitter.

Ainsi, l'objectif de notre projet est donc de proposer un système de classification des Tweets liés au terrorisme en implémentant un programme qui permettra d'analyser et de classer des Tweets selon leurs degrés de radicalité.

Les étapes de la méthode proposée sont résumées dans le schéma suivant (Figure 6) et seront détaillées par la suite.

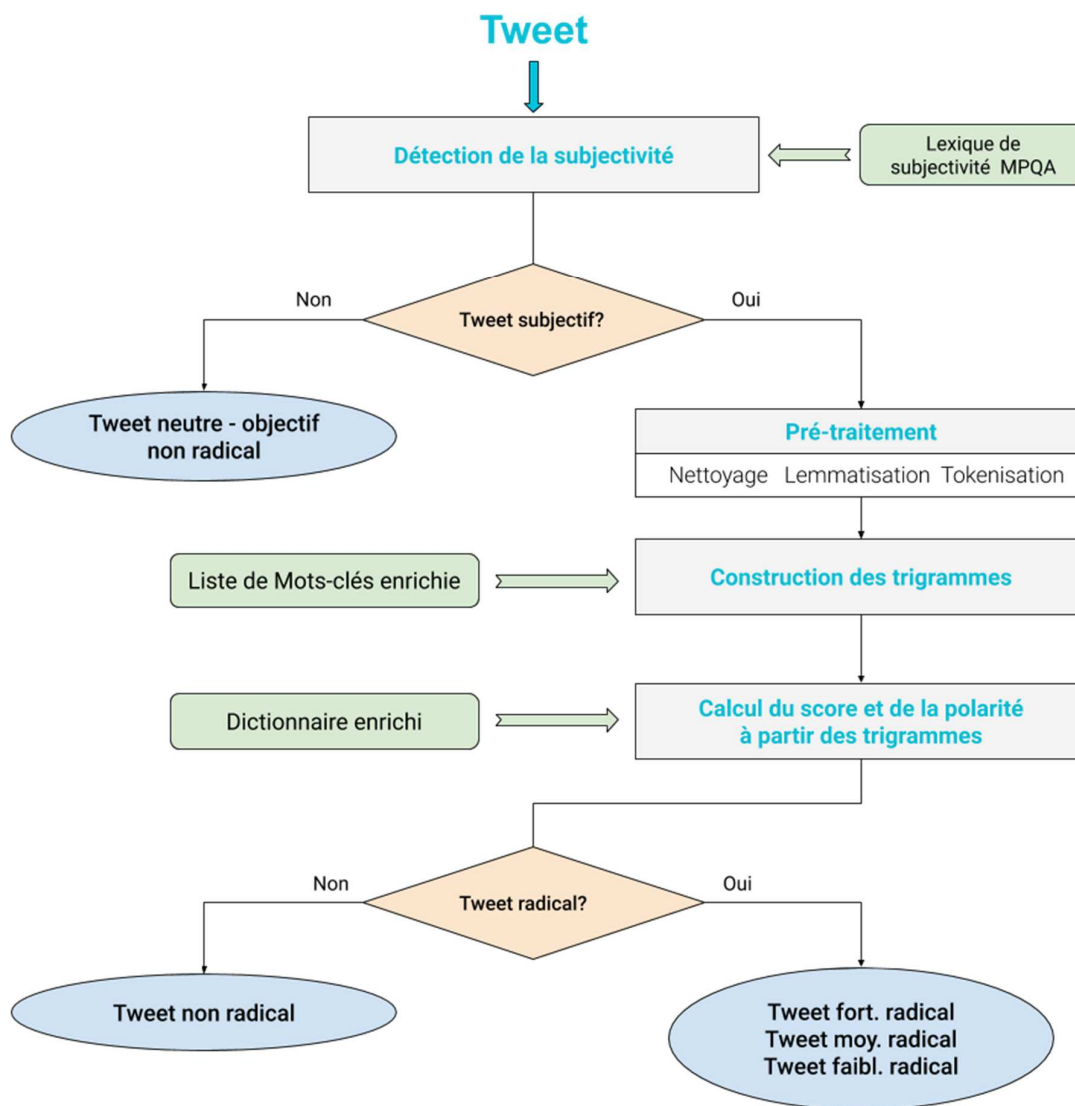


Figure 6 - Schéma résumant les étapes de l'approche proposée.

1. Choix du réseau social

Notre approche est implémentée pour fonctionner sur des données issues du réseau social **Twitter**, bien qu'elle soit applicable à la plupart des médias sociaux en ajustant quelques détails techniques.

Les raisons essentielles motivant ce choix sont que Twitter est la plateforme principale qu'utilisent les groupes terroristes pour y effectuer leurs propagandes, et qu'elle a été également majoritairement choisie par les travaux de recherche qui ont déjà traité ce problème par le passé [22][23][27][29][53].

2. Elaboration et préparation des outils d'analyse

Avant de commencer la phase de l'analyse d'un Tweet, nous devons d'abord préparer les outils suivants, dont nous aurons besoin par la suite : un dictionnaire de termes avec leurs scores de polarité, et une liste de mots-clés de recherche.

Nous avons conçu un modèle Word2vec dans le but d'enrichir les termes du dictionnaire et la liste des mots-clés de recherche pour les rendre plus variés et plus complets.

2.1. Conception du modèle Word2vec

Afin d'enrichir nos outils, nous avons besoin d'un modèle Word2vec adapté à notre contexte. Ainsi, dans cette phase nous avons conçu ce modèle en suivant les étapes résumées dans l'illustration ci-dessous (Figure 7).

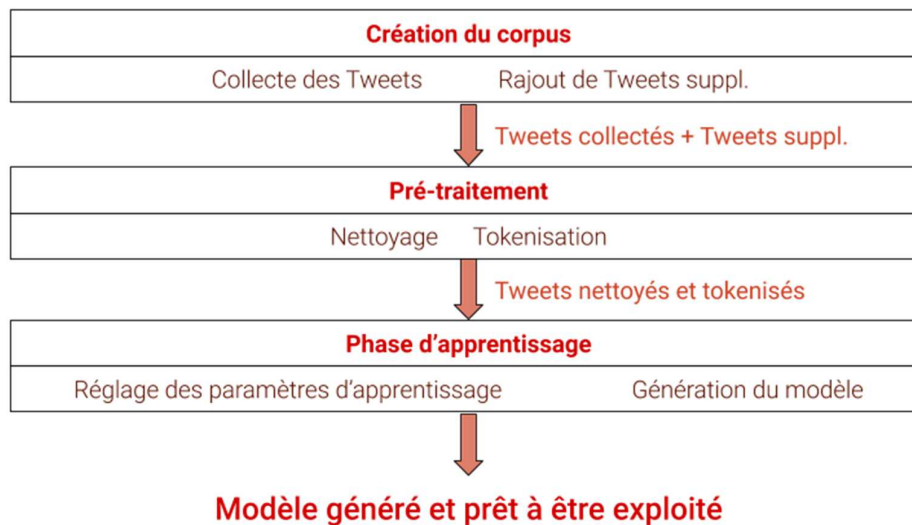


Figure 7 - Schéma résumant les étapes de la conception du modèle Word2vec.

2.1.1. Création du corpus d'apprentissage

i - Collecte des tweets

La première étape de la conception de notre modèle Word2vec consiste à établir un corpus d'apprentissage, dans le contexte de notre domaine. Vu que nous allons utiliser Twitter, nous avons décidé de collecter des Tweets reliés à notre domaine d'étude qu'est le terrorisme.

ii - Nettoyage

Après la collecte, nous avons nettoyé nos Tweets comme suit :

- Mettre le tweet en minuscule.
- Supprimer les URLs.
- Supprimer les #Hashtags.
- Supprimer les emojis.
- Supprimer la ponctuation et les caractères spéciaux.
- Supprimer les numéros.
- Supprimer les stopwords.
- Supprimer les doublons.

2.1.2. Phase d'apprentissage

Une fois le corpus d'apprentissage prêt, vient la phase d'apprentissage du modèle Word2vec. Les paramètres suivants devront être fixés de sorte que le modèle soit le mieux possible :

- ☆ *min_count* : Le nombre minimum d'occurrences qu'un mot doit avoir pour être pris en compte lors de l'apprentissage; les mots dont l'occurrence est inférieure à ce nombre seront ignorés.
- ☆ *size* : La dimension des vecteurs des mots.
- ☆ *workers* : Nombre de threads de travail utilisés pour entraîner le modèle (l'entraînement est plus rapide avec des machines multicœurs).
- ☆ *window* : le nombre de mot, directement lié à un mot X, qui seront pris en considération pour prédire le contexte du mot X.
- ☆ *sg* : L'algorithme d'apprentissage, soit CBOW (0), soit Skip-Gram (1).

Par la suite, l'apprentissage du modèle est lancé sur le corpus créé précédemment.

Après la fin de la phase d'apprentissage, le modèle est généré et prêt à être utilisé.

2.2. Conception du dictionnaire

Pour réaliser notre méthode, nous avons besoin d'un dictionnaire de mots avec leurs polarités, afin de pouvoir calculer le score d'un Tweet donné par la suite.

Pour cela, nous avons choisi un dictionnaire de base existant sur lequel nous avons effectué le prolongement lexical (Word2Vec), afin de le rendre plus riche et plus complet, et aussi plus axé sur notre domaine (voir Figure 8).

Le schéma suivant résume les étapes de la conception du dictionnaire, que nous verrons chacune en détails juste après.

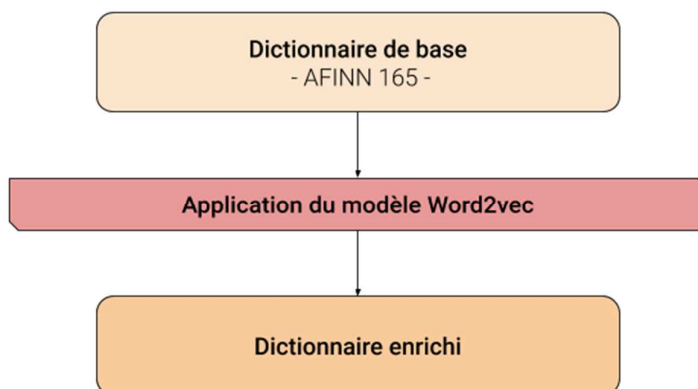


Figure 8 - Illustration qui résume les étapes de la conception du dictionnaire.

2.2.1. Choix du dictionnaire de base

La première phase de la conception de notre dictionnaire, est de choisir un dictionnaire de base.

Nous avons donc choisi le dictionnaire “AFINN”¹, et plus précisément sa dernière version “AFINN-165”² qui contient plus de 3300 mots avec un score de polarité allant de -5, pour les mots les plus négatifs, à +5, pour les mots les plus positifs.

Nous avons choisi ce dictionnaire car il est l'un des dictionnaires les plus simples et les plus populaires. Il a été largement utilisé dans plusieurs travaux traitant de l'analyse des sentiments [44][45][46][47].

¹ <https://github.com/fnielsen/afinn>

² <https://github.com/fnielsen/afinn/blob/master/afinn/data/AFINN-en-165.txt>

2.2.2. Enrichissement du dictionnaire

Une fois le modèle Word2vec généré, nous nous en sommes servis pour enrichir notre dictionnaire de base.

Pour chaque mot du dictionnaire de base, un prolongement lexical est effectué avec le modèle Word2vec, puis on prend les trois mots les plus proches et on les recherche dans le dictionnaire, si un mot n'y existe pas, on le rajoute tout en gardant le même score que le mot initial (Figure 9).

Une fois tous les mots traités, une rapide vérification manuelle est effectuée sur le dictionnaire enrichi, afin de d'y effectuer un nettoyage et de s'assurer de la cohérence des nouveaux mots ajoutés.



Figure 9 - Exemple qui illustre l'étape d'enrichissement du dictionnaire.

2.3. Complétion des mots clés

Afin de construire nos trigrammes, nous devons repérer les mots-clés en relation avec le domaine du terrorisme dans les Tweets que nous allons analyser, et pour cela nous nous sommes servis du modèle Word2vec créé plus tôt pour compléter la liste des mots-clés à partir de quelques mots-clés "graines" (Figure 10).

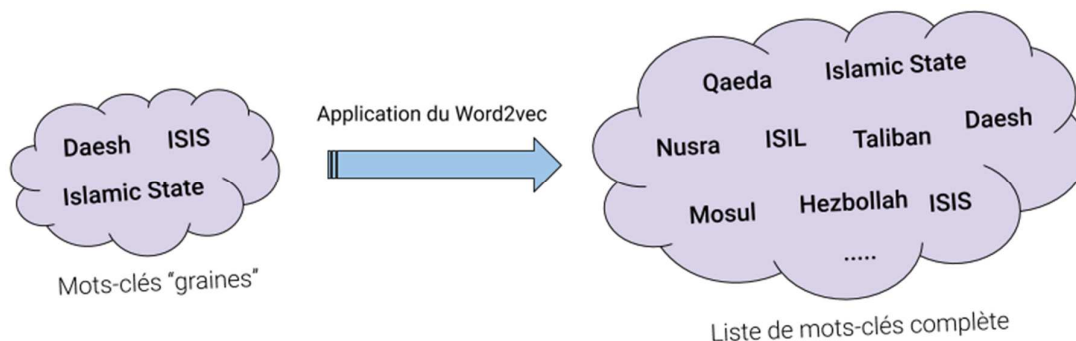


Figure 10 - Exemple du processus de la complétion des mots-clés.

3. Analyse d'un Tweet

Une fois le dictionnaire et la liste de mots-clés prêts, nous sommes désormais disposés à effectuer l'analyse de n'importe quel Tweet donné.

Le processus d'analyse a été divisé en plusieurs phases principales, chaque phase est une phase de préparation à la suivante et elles fonctionnent de manière séquentielle.

3.1. Détection de la subjectivité

La première étape par laquelle passe le Tweet dans notre approche est la détection de subjectivité, le but est de vérifier si le Tweet porte une opinion (Tweet subjectif) avant d'aller plus loin dans l'analyse. Nous nous sommes inspirés de plusieurs travaux effectués précédemment [35][48].

Si un Tweet est objectif (neutre), c'est qu'il ne contient pas de sentiments, et donc pas de radicalité, ainsi, nous n'avons pas besoin de l'analyser plus profondément. Nous pouvons le classer directement comme étant non radical.

Pour effectuer cette détection, nous avons utilisé le lexique de subjectivité MPQA¹.

3.1.1. Lexique de subjectivité MPQA

Le lexique de subjectivité MPQA est une liste de "mots indices de subjectivité", c'est-à-dire des mots qui indiquent généralement l'expression d'une opinion ou d'un avis subjectif, il contient 8222 termes subjectifs. Son but est d'aider à identifier les passages de texte susceptibles de contenir une expression subjective [49]. La figure 11 montre quelques exemples de mots de ce dictionnaire.

WORD	POS	TYPE		WORD	POS	TYPE
able	adj	weaksubj		thank	verb	strongsubj
admirable	anypos	strongsubj		thankful	anypos	strongsubj
attractive	adj	strongsubj		thankfully	anypos	strongsubj
support	noun	strongsubj		thankless	anypos	strongsubj
cute	adj	strongsubj		theoretize	verb	strongsubj
ridiculous	anypos	strongsubj		therefore	anypos	strongsubj
altruist	noun	strongsubj		think	verb	strongsubj
defeat	noun	weaksubj		thinkable	adj	strongsubj
gorgeous	adj	strongsubj		thinking	adj	strongsubj

Figure 11 - Exemples de mots du dictionnaire MPQA.

¹ http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/

3.1.2. Analyse de subjectivité

Si le Tweet ne contient aucun des mots indices présents dans le lexique MPQA, il sera considéré comme étant objectif et neutre, et donc il ne contient aucun sentiment. Ainsi, l'analyse s'arrête ici.

Par contre, si le Tweet contient des mots subjectifs, l'analyse se poursuit avec les étapes ci-dessous, afin de déterminer son sentiment (Figure 12).

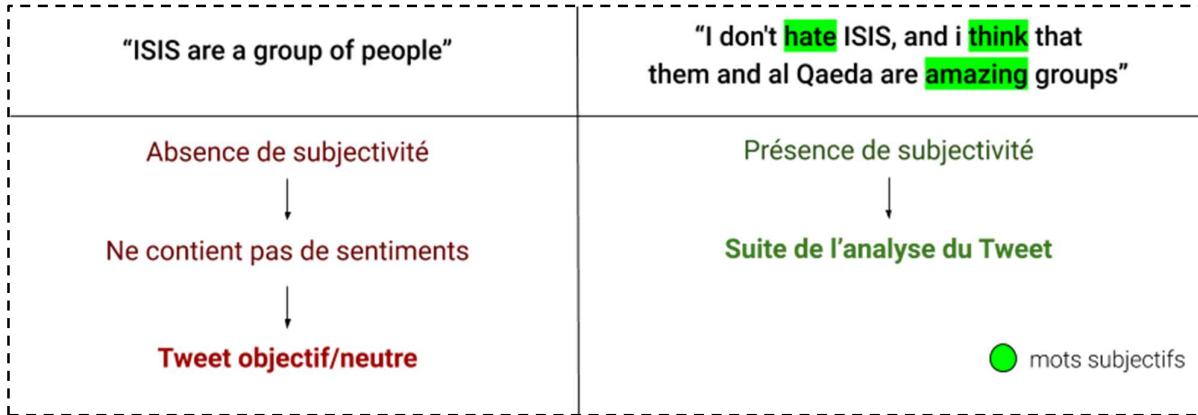


Figure 12 - Exemple de détection de subjectivité.

3.2. Pré-traitement

3.2.1. Nettoyage

Lors de cette étape, nous allons effectuer les traitements suivant au Tweet :

- Mettre le Tweet en minuscule.
- Supprimer les URLs.
- Supprimer les #Hashtags.
- Supprimer les @tags.
- Supprimer les emojis.
- Supprimer la ponctuation et les caractères spéciaux.
- Supprimer les numéros.
- Supprimer les stopwords.
- Remplacer les mots de négation par "!".

3.2.2. Lemmatisation

Le processus de « lemmatisation » consiste à représenter les mots sous leur forme canonique. Par exemple pour un verbe, ce sera son infinitif. Pour un nom, son masculin singulier... L'idée étant de ne conserver que le sens des mots utilisés dans le corpus.

Nous appliquerons donc ce procédé sur le Tweet nettoyé.

3.2.3. Tokenisation

La tokenisation consiste essentiellement à diviser une phrase, un paragraphe ou un document texte entier en unités plus petites, telles que des mots ou des termes individuels. Chacune de ces petites unités est appelée "tokens".

Dans notre cas, nous allons découper le Tweet *nettoyé* et *lemmatisé* en mots, ce seront donc nos tokens.

Toutes les étapes précédentes sont résumées dans la figure suivante (Figure 13).

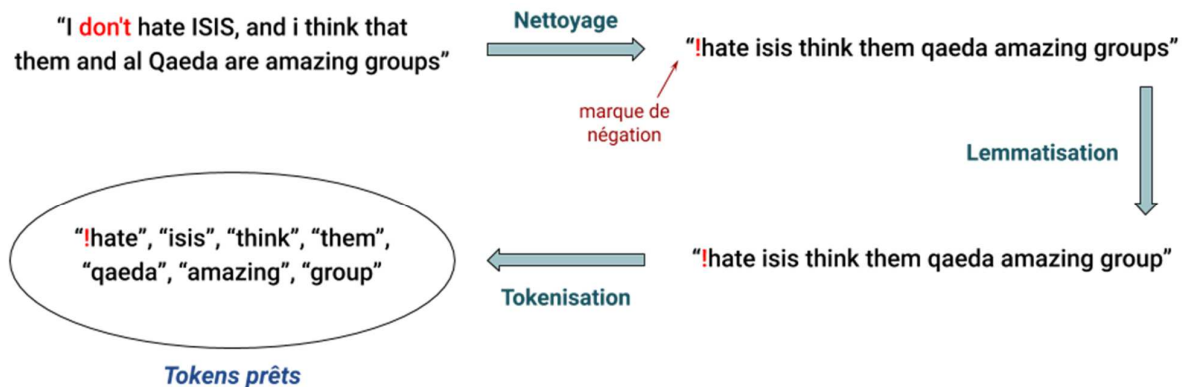


Figure 13 - Exemple de pré-traitement d'un Tweet.

3.3. Création des trigrammes

Lors de cette étape, nous allons former les trigrammes de la forme suivante : mot-clé + terme + terme, ou bien terme + terme + mot-clé.

Pour cela, on repère les mots-clés présents dans le Tweet, et on associe pour chaque mot-clé les deux termes qui le précèdent, et les deux qui le suivent, s'il y en a (voir Figure 14).

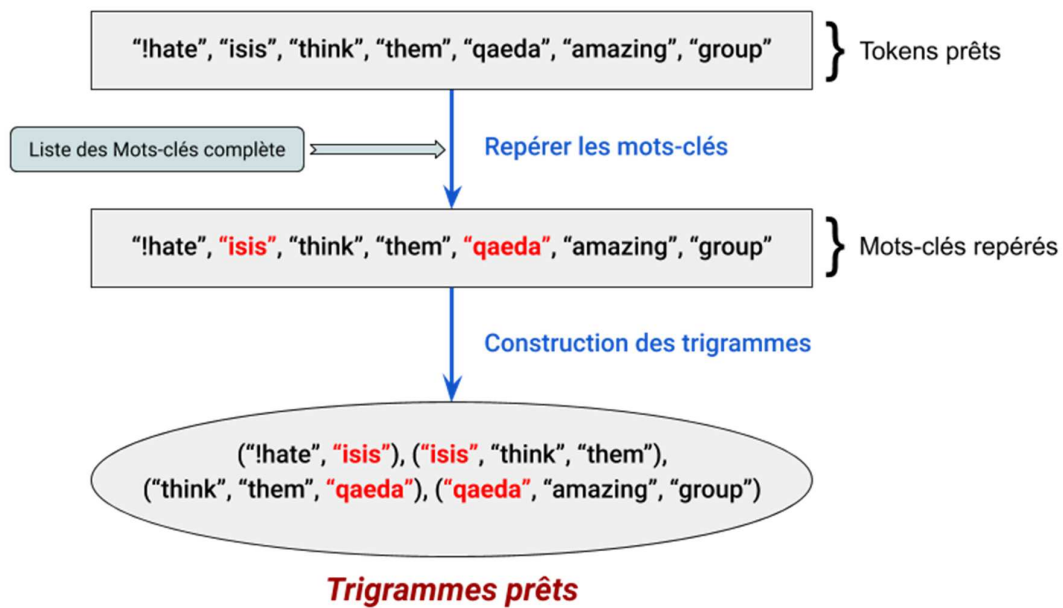


Figure 14 - Exemple de la création des trigrammes.

3.4. Détermination du sentiment

3.4.1. Calcul du score

Pour calculer le score d'un trigramme, on additionne les scores des termes - qu'on récupère depuis le dictionnaire - qui accompagnent les mots-clés qui le composent.

Si le mot n'existe pas dans le dictionnaire, alors on le considère comme étant neutre, et donc son score sera égal à 0.

Dans le cas où un mot est précédé par une négation - '!' -, on multiplie le score du mot $\times -0.5$ [63].

Le score d'un Tweet est donc la moyenne des scores de ses trigrammes non nuls (voir Figure 15).

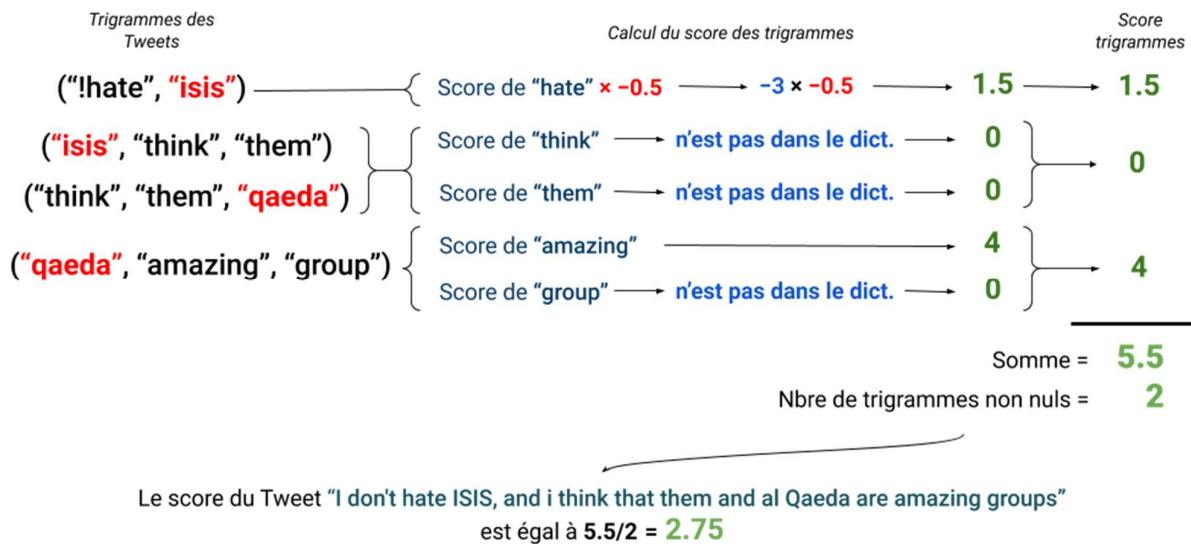


Figure 15 – Exemple de calcul du score d'un Tweet.

3.4.2. Classement de la polarité

Nous avons mis en place un barème (Figure 16) pour classer la polarité d'un tweet selon le score :

- Score inférieur à 0 : Tweet **non radical**.
- Score entre 0 et 2 : Tweet **faiblement radical**.
- Score entre 2 et 4 : Tweet **moyennement radical**.
- Score supérieur à 4 : Tweet **fortement radical**.

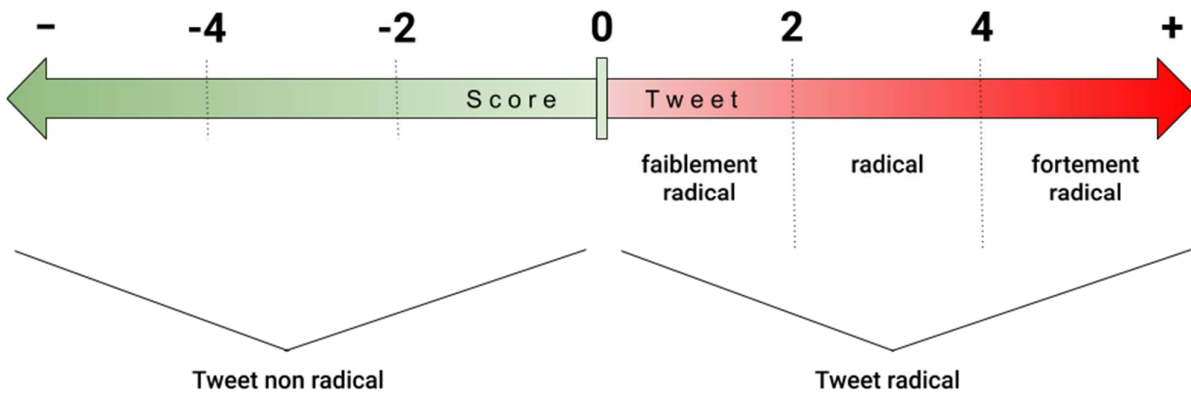


Figure 16 - Barème de classification de Tweets.

CHAPITRE 4

RÉALISATION ET MISE EN ŒUVRE

PARTIE 1 : OUTILS UTILISÉS

1.2. Outils et langage de programmation

Pour la réalisation de notre application, nous avons choisi d'utiliser des outils interopérables et riches en documentation, ce qui nous a facilité leur utilisation.

1.2.1. Python

Python est un langage de programmation interprété et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort et d'une gestion automatique de la mémoire.

Le langage Python est placé sous une licence libre et fonctionne sur la plupart des plates-formes informatiques, des smartphones aux ordinateurs centraux.

Il est également apprécié par certains pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation [50].

Lors de l'implémentation de notre programme, nous avons utilisé plusieurs bibliothèques existantes. Les principaux modules sont brièvement présentés ci-dessous :

1.2.1.1. Gensim

Gensim est une bibliothèque Python pour le topic model¹, l'indexation de documents et la recherche de similitudes dans de grands corpus. Elle est utilisée principalement pour du traitement automatique des langues (NLP) et de l'extraction d'informations (IR) [56].

1.2.1.2. NLTK

Natural Language Toolkit (NLTK) est une bibliothèque Python qui permet l'analyse de texte. Elle fournit des interfaces faciles à utiliser pour plus de 50 corpus et ressources lexicales telles que WordNet, ainsi qu'une suite de bibliothèques de traitement de texte comme pour la classification, la tokenisation et bien plus encore [57].

1.2.1.3. Tweepy

Tweepy est une librairie Python qui permet d'interagir avec l'API Twitter [58].

1.2.2. PyCharm

Développé par l'entreprise tchèque JetBrains, Pycharm est un environnement de développement intégré utilisé pour programmer en Python. Pycharm fonctionne sous Windows, Mac OS X et Linux [51].

Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciel de gestion de versions, et supporte le développement web avec Django [51].

¹ Un **topic model** (modèle thématique ou modèle de sujet) est un modèle probabiliste permettant de déterminer des sujets ou thèmes abstraits dans un document en apprentissage automatique et en traitement automatique du langage naturel.

1.2.3. PyQt

PyQt est un module libre qui permet de lier le langage Python avec la bibliothèque Qt. Il permet ainsi de créer des interfaces graphiques en Python. Une extension de Qt Creator (utilitaire graphique de création d'interfaces Qt) permet de générer le code Python d'interfaces graphiques [52].

1.2.4. L'API Twitter

Twitter offre un service API aux développeurs, elle permet d'accéder aux fonctionnalités de Twitter sans avoir à passer par l'interface du site.

L'API Twitter permet de lire et d'écrire des données Twitter. Ainsi, elle peut être utilisée entre autres pour rédiger des Tweets, parcourir des profils d'utilisateurs, elle permet également d'accéder à un volume élevé de Tweets en recherchant des mots clés spécifiques ou en demandant un échantillon de Tweets publiés par des comptes spécifiques, il est aussi possible de lire des Tweets publiés en temps réel.

L'API Twitter n'est accessible que via des requêtes authentifiées. Pour pouvoir accéder à l'API, Twitter nécessite la création d'un "compte développeur", pour cela il est requis de postuler une demande d'accès en remplissant un formulaire avec des informations sur la future utilisation de l'API, qui sera ensuite examiné afin d'être validé ou rejeté.

Après examination et validation de notre demande par Twitter, nous avons enregistré une nouvelle application sur notre compte développeur, afin de créer et de récupérer nos clés d'authentification, et donc de pouvoir accéder aux services de l'API depuis notre programme.

1.2.5. Twilly - outil construit -

Nous avons constaté que l'API officielle de Twitter a certaines restrictions concernant la collecte, par exemple, nous ne pouvons pas extraire des Tweets qui remontent à plus d'une semaine. Et c'est pour cela que nous avons cherché un autre moyen pour collecter nos Tweets.

Après avoir effectué des recherches, nous avons découvert que Twitter offre la possibilité de faire des recherches avancées de Tweets sur leur site internet¹, sans les limitations de l'API, et en fouillant un peu plus, on a découvert qu'il était possible d'extraire les résultats (Tweets) d'une recherche avancée dans le format JSON².

Nous avons donc construit un outil qui permet d'automatiser tout cela. Nous avons implémenté un programme qui prend en paramètres les filtres de la recherche voulue (exemple dans la Figure 17), puis envoie une requête pour effectuer la recherche, récupère le fichier de résultats JSON, l'analyse pour extraire seulement les informations pertinentes, et crée en sortie un fichier EXCEL (.xlsx) pour faciliter sa manipulation par la suite (voir Figure 18).

¹ <https://twitter.com/search-advanced>

² JSON (JavaScript Object Notation) est un format d'échange de données en texte lisible. Il est utilisé pour représenter des structures de données et des objets simples dans un code qui repose sur un navigateur Web.

Nous avons donné à cet outil le nom de “Twilly”.

```
def collect_twilly(self):
    self.keywords = ["isis", "al qaeda", "suicide bomber", "islamic state", "daech",
                    "terrorism"] # liste de mots-clés à rechercher

    # intervalle de recherche
    self.since = "2019-08-01"
    self.until = "2019-08-07"

    self.maxTweets = 2 # nombre de Tweets à collecter (par mot-clé)
    self.outputFileName = "tweets_exemple.xlsx" # nom du fichier de sortie
    self.lang = 'en' # langue des Tweets à extraire # en -> Anglais
    # self.username = 'nomdutilisateur' # extraire les Tweets d'un utilisateur
```

Figure 17 - Exemple de filtres de recherche.

USERNAME	DATE	RETWEETS	FAVORITES	TEXT	ID	PERMALINK	KEYWORD
johncardillo	2019-08-07 00:59	213	478	- You begged a judge for leniency for nine ISIS	1158890430	https://twitter.com/johncardillo/status/1158890430	isis
KLE1967	2019-08-07 00:59	0	12	White supremacist and white nationalists are	1158890425	https://twitter.com/KLE1967/status/1158890425	isis
KLE1967	2019-08-07 00:59	0	12	White supremacist and white nationalists are	1158890425	https://twitter.com/KLE1967/status/1158890425	al qaeda
masadluffy	2019-08-07 00:59	0	0	We scrutinize anyone with potential links to A	1158890255	https://twitter.com/masadluffy/status/1158890255	al qaeda
SkunkbearAI	2019-08-07 00:49	0	0	You are correct I don't have friends from Syria	1158887885	https://twitter.com/SkunkbearAI/status/1158887885	suicide bomber
navyhm8404	2019-08-07 00:48	0	0	And millions of theists have been killed by oth	1158887706	https://twitter.com/navyhm8404/status/1158887706	suicide bomber
chrisdanne2	2019-08-07 00:59	0	0	The Islamic State is 'resurging' in Syria as US p	1158890352	https://twitter.com/chrisdanne2/status/1158890352	islamic state
AFP	2019-08-07 00:59	30	13	The Islamic State (IS) jihadist group was "resu	1158890245	https://twitter.com/AFP/status/1158890245	islamic state
Mohamedca	2019-08-06 04:16	0	1	Maliki has destroyed Irak. A big racist , à big c	1158577611	https://twitter.com/Mohamedca/status/1158577611	daech
JoeBlac6430	2019-08-05 07:58	0	0	Do you knows why daech don t come in Usa ?	1158270977	https://twitter.com/JoeBlac6430/status/1158270977	daech
AllThingsNat	2019-08-07 00:59	0	4	Do I get to wear a brown shirt. This is the prot	1158890487	https://twitter.com/AllThingsNat/status/1158890487	terrorism
latinos44	2019-08-07 00:59	18	53	We stand united to hold this administrative ac	1158890475	https://twitter.com/latinos44/status/1158890475	terrorism

Figure 18 - Fichier EXCEL obtenu en sortie avec les filtres de recherche précédents.

PARTIE 2 : RÉALISATION DE L'APPROCHE

2.1. Conception du modèle Word2vec

2.1.1 Préparation du corpus

À l'aide de l'outil Twilly, nous avons collecté environ 500 000 Tweets en utilisant des mots-clés de recherche en relation avec le terrorisme. Ces Tweets forment donc notre corpus d'apprentissage.

Nous avons également ajouté à notre corpus environ 800 000 Tweets plus généraux à partir de la base de données Sentiment140¹, afin de rendre notre modèle plus riche et plus varié.

¹ <https://www.kaggle.com/kazanova/sentiment140>

Le corpus subit ensuite un prétraitement, puis, il est prêt à être utilisé.

2.1.2 Phase d'apprentissage

Pour entraîner notre modèle Word2vec, nous avons utilisé la librairie *gensim* de Python.

Avant de lancer l'apprentissage, et après avoir testé plusieurs paramètres différents, nous avons réussi à fixer les paramètres de notre modèle comme suit, car ce sont ceux qui donnaient les meilleurs résultats :

```
model = gensim.models.Word2Vec(min_count=45, size=300, workers=5, window=1, sg=1)
```

Puis nous avons lancé l'apprentissage sur le corpus préparé précédemment avec la ligne suivant :

```
model.train(corpus)
```

Voici quelques résultats obtenus avec le modèle généré :

```
model.most_similar('kill') /* cette fonction donne les dix premiers mots
                             les plus similaires au mot donné */
```

```
>>> [( 'behead', 0.6842024922370911), ('clearers', 0.651875376701355),
      ('killing', 0.6384633779525757), ('killed', 0.635826051235199), ('die',
      0.6354601383209229), ('commit', 0.618017852306366), ('slaughtered',
      0.6136865615844727), ('spies', 0.5997800230979919), ('murdering',
      0.5963218212127686), ('martyred', 0.5803941488265991)]
```

```
model.similarity('isil','alqaida') # calcule la similitude entre deux mots
```

```
>>> 0.9048074
```

2.2. Conception du dictionnaire

Le dictionnaire de base *AFINN-165* contient exactement 3352 termes avec leurs scores correspondants. Après l'application du modèle Word2vec sur chacun des termes précédents, et une vérification manuelle rapide, 1888 nouveaux termes ont été générés.

Le nouveau dictionnaire enrichi contient donc 5240 termes, avec leurs scores allant de -5 à +5 selon leurs degrés de radicalité.

Voici ci-dessous une capture d'écran de quelques mots du dictionnaire (Figure 19), les termes en jaunes sont les termes générés par le Word2vec, ceux en blanc étaient dans le dictionnaire de base.

kidnap	-2	beheads	-3	blaming	-2	bliss	3
massacred	-2	killed	-3	hateful	-2	basking	2
kidnapped	-2	clearers	-3	bless	2	pretty	1
kidnapping	-2	killing	-3	thanking	2	ridiculously	1
terrorizing	-2	behead	-3	love	5	awesomely	3
kill	2	slaughtered	-3	blessing	3	amazingly	3
executed	-3	kills	-3	lifesaver	4	enemy	-2
executes	-3	execute	-3	prayers	1	traitor	-2
executing	-3	kind	2	blind	-1	lose	-3

Figure 19 - Capture d'écran de quelques termes du dictionnaire enrichi.

2.3. Complétion des mots clés

Afin d'élaborer une liste de mots-clés complète adaptée à notre domaine, nous nous sommes servis du modèle Word2vec pour la générer et ce à partir d'une liste de mot-clés de base contenant les mots "ISIS", "Daesh", "AlQaida" & "Terrorism". La portion du code ci-dessous nous a permis de faire cela. Nous avons ensuite enrichi au fur à mesure cette liste à partir des mots obtenus, c'est à dire qu'à chaque fois qu'une suite de mots similaires est générée, nous l'analysons rapidement pour détecter les mots pertinents que nous prendrons dans la liste de mots-clés, et sur lesquels nous effectuerons le même traitement afin d'en générer d'autres. Cette portion du code sera donc appelée de manière récursive pour effectuer ce travail.

```
from gensim.models import KeyedVectors #librairies utilisées

model = KeyedVectors.load("Word2vec.model") # chargement du modèle Word2vec

# fonction qui retourne les mots les plus proches du mot passé en paramètre
sim_words = model.wv.most_similar(mot)
print(sim_words)
```

Code source 1 - Get similar words.

La première exécution sur le mot "ISIS" donne les résultats suivants.

```
>>> [('isil', 0.9359478950500488), ('hamas', 0.9229445457458496), ('alnusra',
0.9163057208061218), ('isisisil', 0.9141430258750916), ('basha', 0.9079852104187012),
('sham', 0.9074201583862305), ('allied', 0.9064512252807617), ('alqaida',
0.9031926393508911), ('moderates', 0.9012107849121094), ('pkk', 0.9009758234024048)]
```

Résultat de l'exécution de la portion de code 1

La liste finale obtenue est présentée ci-dessous.

```
['isis', 'daesh', 'alqaida', 'terrorism', 'isil', 'hezbollah', 'shabaab',  
'alnusra', 'qaida', 'qaeda', 'alqaeda', 'nusra', 'taliban', 'suicidebomber',  
'fsa', 'houthis', 'terrorist', 'sham', 'alsham', 'baghdadi', 'albaghdadi',  
'zawahiri', 'alzawahiri', 'ghaeda', 'alghaeda', 'hizbulla', 'bomber', 'gspc']
```

2.4. Liste des stopwords

Les “stopwords” ou “mots vides” sont des mots qui n’ont pas une réelle signification, c’est-à-dire qu’ils ne portent pas de sens. Les stopwords anglais sont généralement des mots dits grammaticaux tels que des adverbes (actually, always...), des pronoms (i, you...) ou encore des mots de liaison (to, for....).

Durant le nettoyage, lors de la phase de prétraitement, ces stopwords doivent être supprimés afin de faciliter l’analyse. Nous avons établi notre liste en utilisant une liste prédéfinie¹ à laquelle nous avons rajouté d’autres mots vides au fur et à mesure selon nos besoins.

Ci-dessous quelques mots de notre liste, qui contient en tout 1144 mots.

```
"all", "al", "hey", "to", "few", "five", "saturday", "actually", "he", "she",  
"hi", "inner", "is", "while", "set", "ref", "shall", "have", "been", "sub",  
"the", "them", "third", "till", "via", "across", "it", "kg", "cm", "miles",  
"bibi", "november", "nine", "hundred", "here", "are", "him", "behind"
```

2.5. Liste des mots de négation

Les mots de négations sont des mots qui servent à nier, à exprimer le contraire dans une phrase. Dans notre méthode, nous souhaitons traiter ces mots, en les remplaçant durant la phase du prétraitement par le caractère “!”. Il est donc nécessaire d’établir une liste de ces mots de négation :

```
"not", "no", "none", "neither", "never", "nor", "don't", "haven't", "ain't",  
"shouldn't", "musn't", "wouldn't", "didn't", "weren't", "wasn't", "won't",  
"hadn't", "aren't", "isn't", "can't", "couldn't", "cannot", "doesn't",  
"hasn't", "mightn't"
```

¹ https://github.com/sanjaalcorps/EnglishStopWords/blob/master/stop_words_eng.csv

2.6. Quelques exemples de code source

Nous présentons ci-dessous (Code source 2, 3, 4, 5 et 6) quelques parties du code avec lesquelles nous avons réalisés notre approche.

```
def clean_tweet(self, given_tweet):

    tweet = given_tweet.lower()      # mets le Tweet en minuscule
    tweet = re.sub('((www\.[^\s]+)|(https?://[^\s]+)|
        (pic.twitter.com[^\s]+))',' ',tweet)    # supprime les URLs
    tweet = re.sub('@[^\s]+',' ',tweet)          # supprime @username
    tweet = re.sub(r'(?:\s*#[\w]+)+(\s*?)',' ',tweet) # supprime hashatgs
    tweet = demoji.replace(tweet)                # supprime emojis
    tweet = tweet.translate(str.maketrans('', '',
        string.punctuation)) # supprime la ponctuation
    tweet = re.sub('\d+',' ',tweet)              # supprime les nombres
    tweet = remove_stopwords(tweet)              # supprime les stopwords
    tweet = replace_negation_words(tweet)        # traite la négation

    return tweet
```

Code source 2 - Fonction qui nettoie un Tweet.

```
def create_trigrams(self, tokened_tweet): # tokens du Tweet en paramètre
    trigrams = []
    for keyw in keywords:
        if keyw in tokened_tweet:
            # rechercher le ou les indice(s) du mot clé courant
            indices = [i for i, x in enumerate(tokened_tweet) if x == keyw]
            for pos in indices:
                if pos-1 > 0 :
                    left1 = tokened_tweet[pos-1]    # récupérer le mot précédant
                    trigram.append(left1)
                if pos-2 > 0 :
                    left2 = tokened_tweet[pos-2]    # récupérer le 2ème mot précédant
                    trigrams.append(left2)
                if pos+1 <= len(tokened_tweet)-1:
                    right1 = tokened_tweet[pos+1]   # récupérer le mot suivant
                    trigram.append(right1)
                if pos+2 <= len(tokened_tweet)-1:
                    right2 = tokened_tweet[pos+2]   # récupérer le 2ème mot suivant
                    trigram.append(right2)
    return trigrams # retourne une liste des mots formants les trigrammes
```

Code source 3 - Fonction qui forme les trigrammes

```

def score_calculator(self, trigrams_words):
    score = 0
    for word in trigrams_words: # parcours de la liste des mots des trigrammes
        if '!' in word : # vérifie si le mot est précédé d'une négation
            mot=re.sub(r'!', "", word)
            if mot in dict.keys(): # vérifie l'existence du mot dans le dict.
                # mot précédé d'une négation -> score multiplié x -0.5
                score = score + int(dict[mot]) * (-0.5)
        else:
            if big in dict.keys(): # vérifie l'existence du mot dans le dict.
                score = score + int(dict[big])
    return score

```

Code source 4 - Fonction qui calcule le score à partir d'une liste de mots

```

def analyse_trigrams(self, tweet):

    subjectivity = detect_subj(tweet)
    # retourne Vrai si le Tweet est subjectif, sinon retourne Faux

    if subjectivity == True:
        # Tweet subjectif
        clean_tweet = pretreatment(tweet)           # nettoie le Tweet
        tokens = tokenization(clean_tweet)           # tokenise le Tweet
        trigrams= create_trigrams(tokens)            # forme les trigrammes
        score = score_calculator(trigrams)           # calcule le score
        polarity = classification(score)              # classifie le score du Tweet
    else:
        # Tweet non subjectif
        score = 0
        polarity = classification(score)              # classifie le score du Tweet

    return [tweet, score, polarity]

```

Code source 5 - Fonction qui analyse un Tweet

PARTIE 3 : FONCTIONNALITÉS ADDITIONNELLES

Une fois notre méthode implémentée, nous avons réfléchi pour trouver des manières pertinentes et utiles de l'appliquer. Nous avons donc pensé aux fonctionnalités présentées dans ce qui suit :

3.1. Examen de l'historique d'un utilisateur

La première fonctionnalité est l'examen de l'historique d'un utilisateur donné afin de déterminer si celui-ci publie du contenu radical de manière fréquente, de temps à autre, ou pas du tout.

La première phase de l'examen consiste à récolter les Tweets contenant des mots-clés depuis le profil de l'utilisateur. Puis, les Tweets récoltés sont analysés, et les résultats suivants sont retranscrits :

- Le nombre de Tweets contenant un mot-clé de recherche.
- Le nombre de Tweets faiblement radicaux.
- Le nombre de Tweets moyennement radicaux.
- Le nombre de Tweets fortement radicaux.
- Le nombre de Tweets non radicaux.

3.2. Collecte de Tweets en temps réel

Une autre idée que nous avons mise en œuvre est d'analyser les Tweets publiés en temps réel, et ce grâce à l'API Twitter, combinée à l'examen de l'historique d'utilisateurs lorsque nécessaire.

L'API Twitter propose une fonction "streaming" de Tweets, celle-ci permet de recevoir et rester à l'écoute en temps réel des derniers Tweets publiés. Nous avons utilisé la liste des mots-clés relatifs au terrorisme que nous avons préparé précédemment afin de ne filtrer et de n'extraire que les Tweets contenant au moins l'un de ces mots. Lorsqu'on lance le programme, l'écoute commence et reste en attente de l'arrivée d'un Tweet.

Lorsqu'un Tweet arrive, on lui effectue l'analyse avec notre méthode. Si le Tweet n'est pas radical, on repasse à l'écoute, mais si par contre le Tweet est radical, une examen de l'historique de l'utilisateur auteur est effectuée, et les résultats sont retranscrits.

Le schéma ci-dessous (Figure 20) illustre le fonctionnement de cette fonctionnalité.

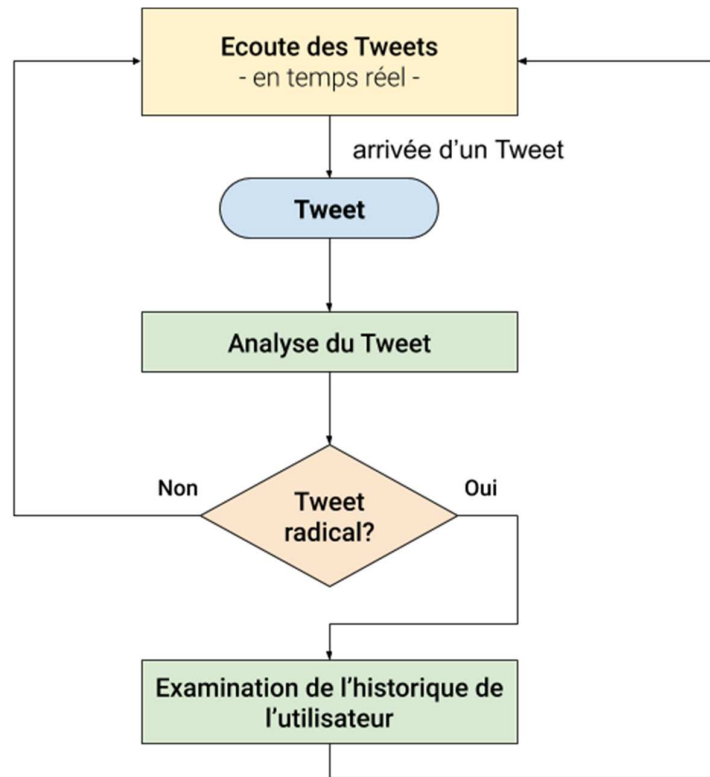


Figure 20 - Illustration du fonctionnement de la fonctionnalité "Temps réel".

3.3. Interface graphique

Nous avons également créé une interface graphique qui regroupe les fonctionnalités précédentes, grâce à la librairie *PyQT*, afin de rendre plus facile leurs manipulations par les utilisateurs et faciliter aussi la visualisation des résultats.

Voici ci-dessous quelques captures d'écran de cette interface :



Figure 21 - Capture d'écran de l'interface graphique.

Sur la première capture d'écran - *Figure 21* -, l'interface présentée permet à l'utilisateur de choisir entre lancer une analyse en temps réel, ou bien de lancer l'examen de l'historique d'un utilisateur de Twitter, en entrant son nom d'utilisateur.

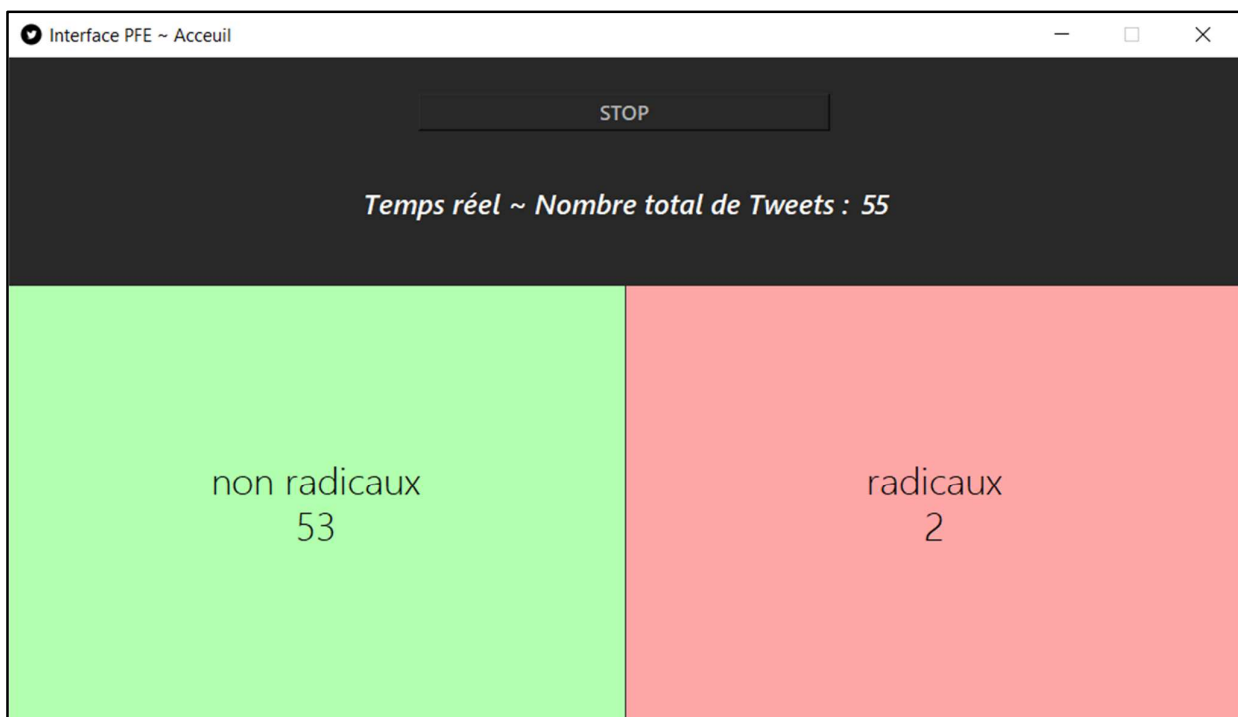
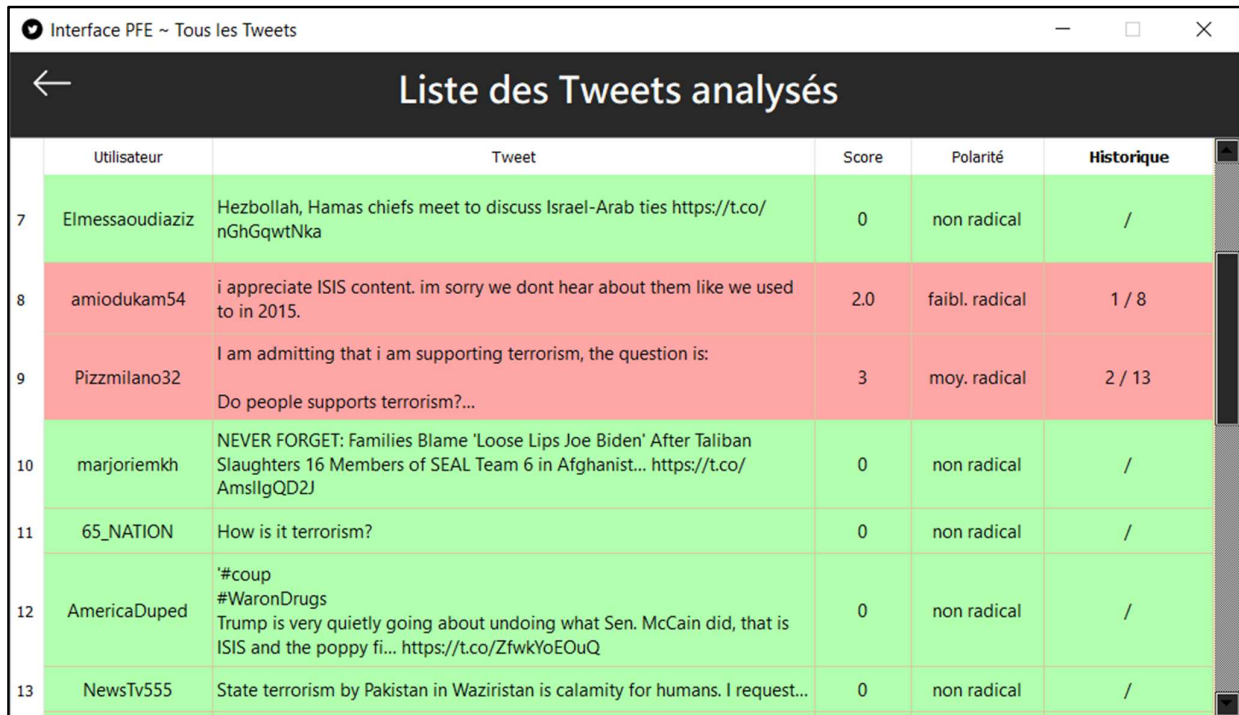


Figure 22 - Capture d'écran de l'interface graphique pendant une exécution.

Cette deuxième capture d'écran - Figure 22 - a été effectuée pendant une exécution de l'analyse en temps réel, l'interface se met à jour à chaque fois qu'un Tweet arrive et est analysé. Les boutons vert et rouge permettent d'accéder à la liste détaillée des Tweets de chaque classe comme présentée dans la capture d'écran de la Figure 23.



	Utilisateur	Tweet	Score	Polarité	Historique
7	Elmessaoudiaziz	Hezbollah, Hamas chiefs meet to discuss Israel-Arab ties https://t.co/nGhGqwtNka	0	non radical	/
8	amiodukam54	i appreciate ISIS content. im sorry we dont hear about them like we used to in 2015.	2.0	faibl. radical	1 / 8
9	Pizzmilano32	I am admitting that i am supporting terrorism, the question is: Do people supports terrorism?...	3	moy. radical	2 / 13
10	marjoriemkh	NEVER FORGET: Families Blame 'Loose Lips Joe Biden' After Taliban Slaughters 16 Members of SEAL Team 6 in Afghanist... https://t.co/AmsllgQD2J	0	non radical	/
11	65_NATION	How is it terrorism?	0	non radical	/
12	AmericaDuped	#coup #WaronDrugs Trump is very quietly going about undoing what Sen. McCain did, that is ISIS and the poppy fi... https://t.co/ZfwkYoEOuQ	0	non radical	/
13	NewsTv555	State terrorism by Pakistan in Waziristan is calamity for humans. I request...	0	non radical	/

Figure 23 - Capture d'écran des Tweets analysés sur l'interface graphique pendant l'exécution précédente.

Note :

- La colonne "Tweet" représente le Tweet arrivée en temps réel.
- La colonne "Utilisateur" contient l'auteur de chaque Tweet.
- La colonne "Score" représente le score de radicalité de ce Tweet.
- La colonne "Polarité" représente la classe à laquelle a été classé chaque Tweet.
- La colonne "Historique" représente le nombre de Tweets radicaux du profil de l'utilisateur / nombre de Tweets contenant un mot-clé du profil de l'utilisateur.

CHAPITRE 5

ÉVALUATION ET DISCUSSION

Afin d'évaluer les performances de notre approche et d'exposer les pertinences de l'utilisation d'un bon modèle Word2Vec pour enrichir le dictionnaire, nous avons comparé l'approche proposée à d'autres approches déjà existantes, nous avons donc pris deux méthodes parmi les plus connues qui sont :

- La méthode "classique" : le terme "classique" est une référence à la façon basique qui a été utilisée pour effectuer le traitement avec cette méthode. Elle consiste à :
 - Effectuer le prétraitement du Tweet.
 - Tokeniser le Tweet et rechercher chaque token dans le dictionnaire.
 - Additionner les scores des mots trouvés dans le dictionnaire.
 - Classer le Tweet selon le score obtenu.
- La méthode utilisant les bigrammes : cette méthode partage le même principe que la méthode que nous avons proposée. Au lieu de s'intéresser aux trigrammes qui entourent les mots-clés, cette méthode s'intéresse plutôt aux bigrammes.

Nous avons évalué ces méthodes ainsi que la nôtre d'abord avec le dictionnaire de base - dict. *AFINN-165* - et ensuite avec le dictionnaire enrichi, afin d'évaluer l'impact de l'enrichissement avec le Word2vec.

Pour effectuer l'évaluation, nous avons constitué un échantillon d'environ 200 Tweets (Tableau 2) que nous avons annoté manuellement.

		Radicaux			Non radical
		Faibl.	Moy.	Fort.	
Nombre de Tweets	13	19	20	148	
	52				

Tableau 2 - Statistiques des Tweets annotés manuellement.

Nous avons ensuite testé chaque méthode avec chaque dictionnaire sur cet échantillon afin de calculer leurs performances et ce en utilisant les trois formules suivantes :

- La précision : La précision permet de répondre à la question : *“Quelle proportion d'identifications positives était effectivement correcte ?”*

La précision d'une classe i évalue la proportion de Tweets correctement attribués par le nombre de Tweets attribué par la méthode à la même classe. La précision est de 1 lorsque tous les Tweets classés par le système dans la classe i sont corrects et 0 si aucun des Tweets renvoyés n'est bien classé. Le calcul de la précision s'effectue avec l'équation suivante :

$$\text{Précision} = \frac{\text{nb Tweets correctement attribués à la classe } i}{\text{nb total de Tweets attribués à la classe } i}$$

- Rappel : Le rappel permet quant à lui de répondre à la question suivante : *“Quelle proportion de résultats positifs réels a été identifiée correctement ?”*

Le rappel d'une classe i évalue la proportion de Tweets correctement attribués par le nombre de Tweets qui appartiennent réellement à la même classe. Si le rappel vaut 1, c'est que les tweets pertinents disponibles ont tous été correctement classés par le système, à l'inverse, si le rappel est 0, c'est qu'aucun Tweet de la classe n'a été correctement classé. Le calcul du rappel se fait avec l'équation suivante :

$$\text{Rappel} = \frac{\text{nb Tweets correctement attribués à la classe } i}{\text{nb total de Tweets appartenant à la classe } i}$$

- Mesure F : La F-mesure ou Mesure F permet de calculer une valeur qui synthétise les deux mesures vues plus tôt. Elle s'effectue en calculant la moyenne harmonique de la précision et du rappel en utilisant l'équation suivante :

$$F = 2 \times \frac{\text{précision} \times \text{rappel}}{\text{précision} + \text{rappel}}$$

Dans un premier temps, nous allons évaluer et comparer les méthodes par rapport à leurs capacités à détecter et différencier les Tweets radicaux des Tweets non radicaux. Puis par la suite, nous évaluerons leurs performances dans la détermination du degré de radicalité dans les Tweets radicaux. Nous effectuons aussi une analyse automatique sur 200,000 Tweets avec les trois méthodes : trigrammes, bigrammes et classique.

1. PARTIE 1 - DÉTECTION DE RADICALITÉ

Le tableau suivant (Tableau 3) résume les résultats de l'évaluation de la détection des Tweets radicaux des non radicaux avec les trois méthodes comme expliqué précédemment. Les résultats F-mesure sont représentés par un graphe de colonne dans la Figure 24.

Méthode	Dictionnaire utilisé	Classes	Nb Tweets classés	Précision	Rappel	F-Mesure
Méthode Proposée	Dictionnaire non enrichi	Radical Non Radical Moyenne	49 151 -	0.63 0.86 0.75	0.60 0.88 0.74	0.61 0.87 0.75
	Dictionnaire enrichi	Radical Non Radical Moyenne	65 135 -	0.65 0.92 0.78	0.79 0.84 0.81	0.70 0.88 0.79
Méthode Bigrammes	Dictionnaire non enrichi	Radical Non Radical Moyenne	55 145 -	0.63 0.87 0.75	0.67 0.86 0.73	0.65 0.86 0.73
	Dictionnaire enrichi	Radical Non Radical Moyenne	67 133 -	0.64 0.90 0.76	0.83 0.82 0.82	0.71 0.86 0.77
Méthode Classique	Dictionnaire non enrichi	Radical Non Radical Moyenne	53 147 -	0.55 0.84 0.70	0.56 0.84 0.70	0.55 0.84 0.70
	Dictionnaire enrichi	Radical Non Radical Moyenne	74 126 -	0.54 0.90 0.72	0.77 0.77 0.77	0.63 0.83 0.73

Tableau 3 - Résultats de l'évaluation de la détection de radicalité.

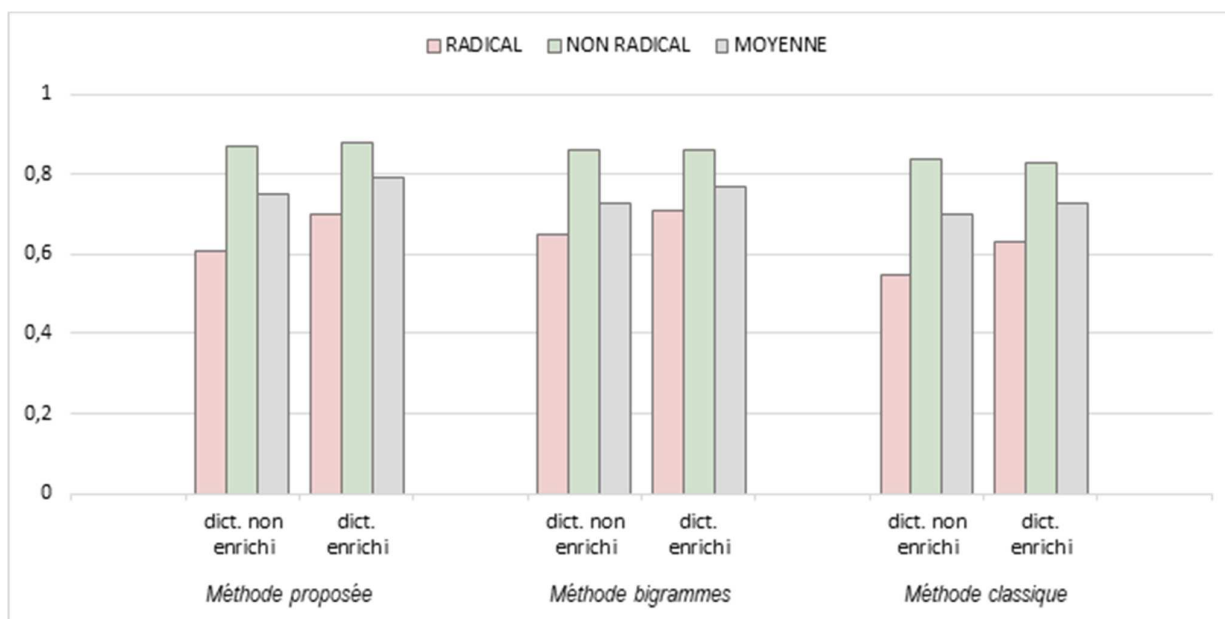


Figure 24 - Graphe de colonnes regroupant les résultats F-mesure de l'évaluation de la détection du contenu radical.

2. PARTIE 2 - CATÉGORISATION DES DEGRÉS DE RADICALITÉ

Dans cette partie, nous avons évalué les approches précédentes sur leurs capacités à correctement classer les degrés de radicalité des Tweets radicaux. Les résultats sont regroupés dans le tableau suivant (Tableau 4) et représentés par un graphe de colonne dans la figure 25.

Méthode	Dictionnaire utilisé	Classes	Nb Tweets classés	Précision	Rappel	F-Mesure
Méthode proposée	Dictionnaire non enrichi	Faibl. radical	15	0.71	0.38	0.50
		Moy. radical	32	0.35	0.44	0.39
		Fort. radical	2	1.00	0.05	0.10
		Moyenne	-	0.67	0.29	0.33
Méthode proposée	Dictionnaire enrichi	Faibl. radical	23	0.50	0.46	0.48
		Moy. radical	35	0.43	0.56	0.49
		Fort. radical	7	1.00	0.30	0.46
		Moyenne	-	0.64	0.44	0.48
Méthode bigrammes	Dictionnaire non enrichi	Faibl. radical	16	0.56	0.38	0.45
		Moy. radical	38	0.27	0.39	0.32
		Fort. radical	1	0.00	0.00	0.00
		Moyenne	-	0.27	0.26	0.26
Méthode bigrammes	Dictionnaire enrichi	Faibl. radical	20	0.46	0.46	0.46
		Moy. radical	36	0.38	0.44	0.41
		Fort. radical	11	0.89	0.40	0.55
		Moyenne	-	0.58	0.43	0.46
Méthode classique	Dictionnaire non enrichi	Faibl. radical	31	0.40	0.46	0.43
		Moy. radical	21	0.43	0.33	0.38
		Fort. radical	1	0.00	0.00	0.00
		Moyenne	-	0.28	0.26	0.27
Méthode classique	Dictionnaire enrichi	Faibl. radical	47	0.29	0.54	0.38
		Moy. radical	21	0.58	0.39	0.47
		Fort. radical	6	1.00	0.22	0.33
		Moyenne	-	0.62	0.38	0.39

Tableau 4 - Résultats de l'évaluation sur les degrés de radicalité.

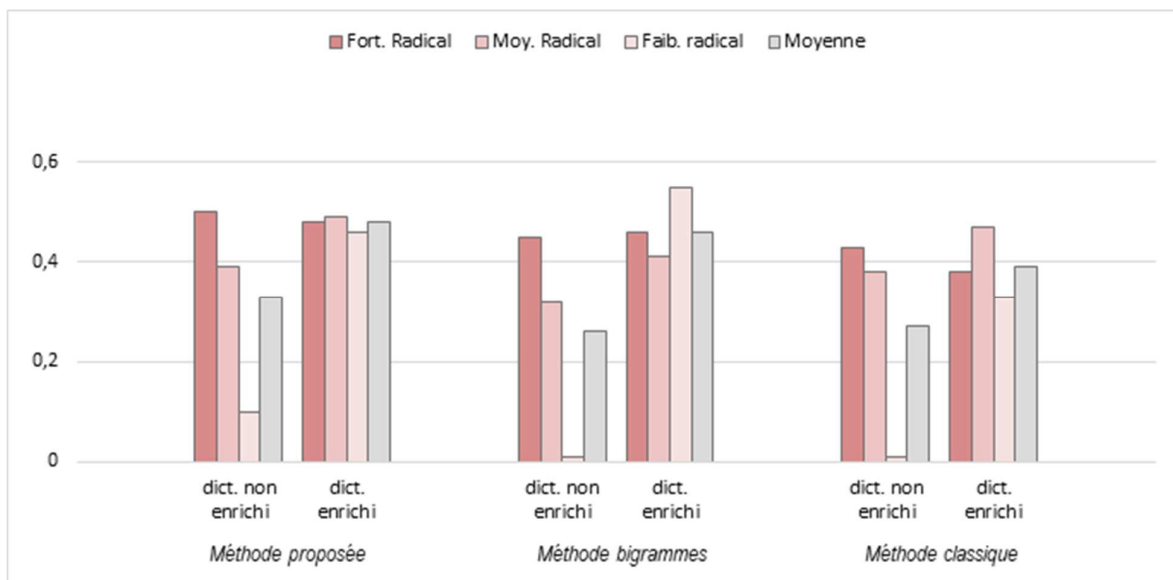


Figure 25 - Graphe de colonnes des résultats F-mesure de l'évaluation sur le degré de radicalité.

3. PARTIE 3 - TEST SUR UN GRAND NOMBRE DE TWEETS

Nous avons testé la méthode proposée ainsi que les deux autres méthodes sur environ 200,000 Tweets (automatiquement) que nous avons collectés. Le tableau (Tableau 6) regroupe les résultats obtenus qui sont représentées en graphe dans la figure 26.

	Méthode classique		Méthode bigrammes		Méthode proposée	
	Dict. de base	Dict. enrichi	Dict. de base	Dict. enrichi	Dict. de base	Dict. enrichi
Fort. radical	0.23% (≈456)	0.29% (≈575)	0.11% (≈218)	0.21% (≈416)	0.24% (≈476)	0.94% (≈1863)
Moy. radical	5.23% (≈10,363)	15.13% (≈29,981)	3.56% (≈7,054)	21.09% (≈41,791)	4.26% (≈8,441)	19.07% (≈37,789)
Faib. radical	8.78% (≈17,398)	18.93% (≈37,510)	3.02% (≈5,984)	6.61% (≈13,098)	4.26% (≈8,441)	9.27% (≈18,368)
Non radical	85.76% (≈169,937)	65.65% (≈130,088)	93.31% (≈184,898)	72.09% (≈142,849)	91.24% (≈180,796)	70.72% (≈140,134)

Tableau 6 - Résultats de classification des Tweets par trois méthodes.

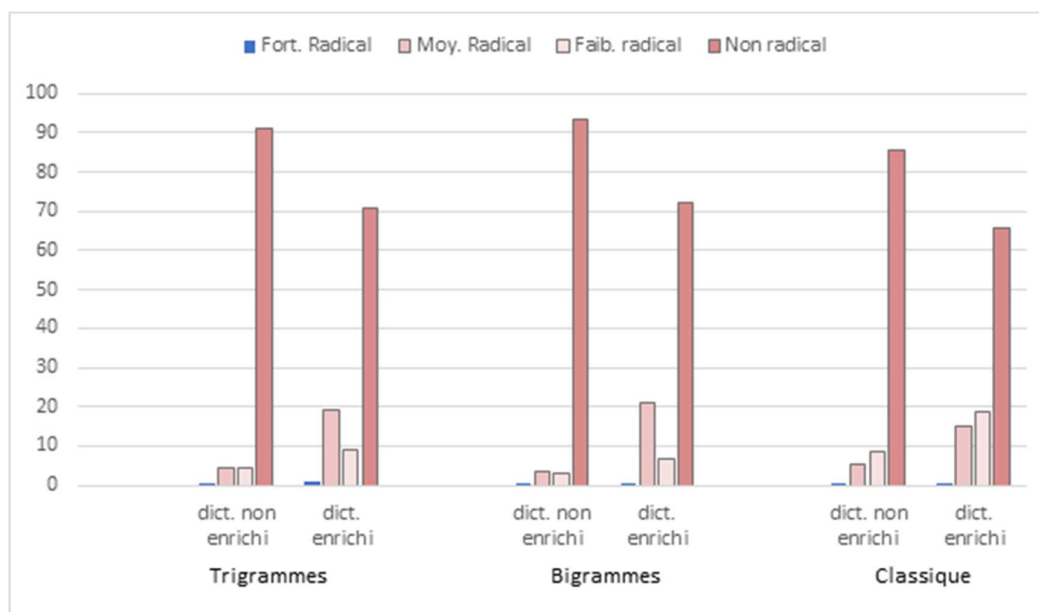


Figure 26 - Graphe de colonne regroupant les résultats des Tweets trié par les trois méthodes.

4. DISCUSSION DES RÉSULTATS

Sur l'évaluation de la détection de la radicalité, on constate que la meilleure performance est fournie par la méthode proposée des trigrammes (0.79 F-Mesure en utilisant le dictionnaire enrichi), vient ensuite la méthode des Bigrammes avec sa F-Mesure qui équivaut à 0.77. Puis, vient la méthode classique qui est la moins performante avec 0.73 de F-Mesure.

On constate également de ces résultats que la méthode proposée classe les Tweets non-radicaux avec une précision très élevée (0.88 de F-Mesure), ainsi que les deux autres méthodes à une mesure légèrement moindre (0.86 pour la méthode des Bigrammes, 0.83 pour la méthode classique).

On remarque aussi sur le *Tableau 3* que les résultats obtenus avec les trois méthodes sont significativement meilleurs lorsqu'elles ont été appliquées avec le dictionnaire enrichi plutôt qu'avec le dictionnaire de base (F-Mesure qui passe de 0.75 avec le dict. de base à 0.79 avec le dict. enrichi pour la méthode proposée, de 0.73 à 0.77 pour la méthode des Bigrammes et de 0.70 à 0.73 pour la méthode classique).

Sur l'évaluation de la catégorisation des degrés de radicalité, on relève à première vue que là aussi les résultats des méthodes avec le dictionnaire enrichi sont bien meilleurs que ceux avec le dictionnaire de base (0.33 à 0.48 pour la méthode proposée, 0.26 à 0.46 pour la méthode des Bigrammes et 0.27 à 0.39 pour la classique). Ceci prouve donc que l'utilisation du Word2vec a amené un apport éloquent aux performances de ces méthodes.

On remarque tout de même que les résultats ne sont pas élevés, la méthode proposée possède la meilleure efficacité avec un F-Mesure de 0.48, la méthode des bigrammes vient ensuite avec 0.46, et puis 0.39 pour la méthode classique. Cela est dû notamment au peu de données d'évaluation disponibles, il y aurait peut-être de meilleurs résultats avec plus de données ou des approches de classification plus complexes.

D'après le tableau 6, on remarque que le pourcentage des Tweets classés non radicaux se dégrade pour les trois méthodes en utilisant le dictionnaire enrichi (de 85.76% à 93.31%, de 91.24% à 65.95% et de 72.09% à 70.72% pour la méthode classique, bigrammes et trigrammes resp.), pour augmenter en parallèle pour les trois autres classes (légèrement pour la classe fortement radicale et d'une amélioration de 10% pour les deux classes restantes. Ceci prouve encore une fois que l'utilisation du Word2Vec, qui permet d'insérer de nouveaux mots d'opinions dans le dictionnaire, apporte des améliorations importantes.

5. EXEMPLES DE SORTIE

Le tableau suivant -Tableau 5- regroupe quelques exemples de Tweets annoter manuellement et par l'approche proposée.

Exemple	Tweet	Annotation manuelle	Annotation de l'approche
1	Terrorism is great. I love it.	fortement radical	fortement radical
2	isis and daesh are doing a great job by killing all those bad people	fortement radical	fortement radical
3	When people believe in god, nothing else matters. I support al qaeda, ISIS, and any organization eradicating the scum the pollute the earth	moy. radical	moy. radical
4	I support DAESH . Support our troops.	moy. radical	moy. radical
5	I enjoy ISIS videos.	faiblement radical	faiblement radical
6	I'm a Muslim. I'm an America and I don't support terrorism.	non radical	non radical
7	No, They use the name of Allah to practice their dirty religion in the name of Allah, We are against them and they are never represent us, they only represent themselves. We hate ISIS just as much as you do, and ISIS has nothing to do with Islam and they never represents it	non radical	non radical
8	I pray that the world discovers peace and the light of love. Terrorism is a danger to the humanity. We must fight it with all our might and will.	non radical	faiblement radical
9	Groups like ISIS are located in the middle east	non radical	moy. radical
10	I appreciate deash very much	fortement radical	moy. radical

Tableau 5 - Exemples de sortie.

Certaines erreurs se sont glissées et sont expliquées dans ce qui suit :

- Exemple 8 : L'erreur est liée à la non prise en charge de la ponctuation par l'approche. Elle est causée par la présence du mot positif *"love"* près du mot clé *"terrorism"* dans le trigramme, et ce malgré la présence du mot négatif *"danger"* dans le deuxième trigramme car le poids de ce dernier n'est pas suffisant. Une perspective d'amélioration pourrait être de trouver un moyen de gérer la ponctuation et la structure des Tweets.
- Exemple 9 : L'erreur de classification est due ici au fait que le programme considère le mot *"like"* ayant le sens *"aimer"*, alors que dans le contexte du Tweet, il exprime la conjonction *"comme"*. Cet exemple relève que l'approche ne considère pas le contexte des Tweets analysés, ce qui peut causer des erreurs dans certains cas. Donc, prendre en compte le contexte des mots des Tweets peut être une optique d'amélioration.

- Exemple 10 : Dans cet exemple, la présence de “*very much*” accentue plus l’intensité du mot “*appreciate*”, ce qui nous a amené à le classer comme étant fortement radical, mais notre approche ne prend pas en compte ces cas ci, ce qui peut causer des erreurs dans la prédiction du degré de radicalité. Considérer ces mots d’intensité pourrait améliorer les résultats.

6. SYNTHÈSE

Dans ce chapitre, nous avons évalué l’approche proposée en la comparant à deux autres méthodes : la méthode classique et la méthode des bigrammes, et ce sur un dataset contenant environ 200 Tweets que nous avons collecté et annoté manuellement. Nous avons également testé les méthodes précédentes sur un très grand nombre de Tweets.

Par la suite, nous avons discuté des résultats de cette évaluation et des renseignements à en tirer. Puis, nous avons abordé des exemples de sorties, et expliqué quelques erreurs qui peuvent occurer.

Nous concluons de ce chapitre que notre approche donne de bons résultats comparativement aux autres méthodes. Elle est plutôt performante dans la détection de la radicalité, et est globalement moyenne dans la catégorisation du degré de radicalité. Il existe des points d’améliorations à apporter dans le futur afin d’améliorer davantage son efficacité.

7. PERSPECTIVES

Pour des futures améliorations nous proposons donc les perspectives suivantes :

- Affiner le dictionnaire et l’enrichir avec des termes encore plus spécifiques au domaine étudié.
- Elargir les données disponibles afin d’avoir des résultats plus précis.
- Intégrer d’autres techniques de Machine Learning.
- Utiliser les quadrigrammes ou d’autres n-grammes.
- Prendre en considération le contexte et le sens des mots dans la phrase.
- Utiliser d’autres paramètres comme les emojis, les signes de ponctuations ou les mots d’intensité pour mieux distinguer le sarcasme, l’ironie, l’humour, l’intensité ou même le sens caché.
- Appliquer la logique floue pour mieux distinguer et catégoriser les classes de polarités.

CONCLUSION GÉNÉRALE

L'objectif de ce mémoire est de proposer un système de classification de Tweets rédigés en anglais liés au terrorisme en implémentant un programme qui permettra d'analyser et de classer des Tweets selon leurs degrés de radicalité.

Pour réaliser cela, nous avons proposé une approche lexicale utilisant un dictionnaire enrichi par un modèle Word2Vec, basée sur les trigrammes.

Pour présenter le sujet, nous avons commencé par donner quelques définitions de base, présenter la plateforme Twitter et par réaliser un état de l'art sur l'analyse des sentiments où nous avons étudié quelques travaux existants liés au terrorisme. Nous avons ensuite présenté les outils utilisés pour l'analyse : l'outil Twilly pour la collecte des Tweets, les étapes de prétraitement, la construction des trigrammes et l'attribution des scores. Quelques portions des codes implémentés pour ces phases ont été présentées par la suite.

Ce travail offre une comparaison à deux autres méthodes couramment utilisées pour évaluer l'efficacité de l'approche proposée à la détection du contenu radical et de la catégorisation des degrés de radicalité, et ce sur une centaine de données annotées manuellement. Nous avons obtenu de bon résultats (81% F-mesure pour la détection du contenu radical et 48% pour détecter correctement le degré de radicalité).

Afin d'aboutir à une réalisation un peu plus concrète de l'approche proposée, nous avons implémenté un programme qui permet de traquer et d'analyser l'historique des comptes des utilisateurs. Et nous avons également mis en place notre système pour analyser les Tweets en temps réel, grâce à l'API Twitter. De plus, nous avons réalisé une interface graphique pour faciliter la manipulation et améliorer la visualisation des résultats.

La partie la plus difficile du travail a été de regrouper un bon nombre de Tweets radicaux vu que les groupes terroristes s'expriment beaucoup plus en langue arabe. Cela est due aussi au fait que Twitter donne une grande importance à la détection de la propagande terroriste sur sa plateforme, donc analyse constamment les comptes des utilisateurs et les suspend, ainsi les contenus radicaux sont rapidement supprimés.

La principale contribution de ce projet a été d'utiliser les trigrammes liés directement aux mots clés, ainsi que l'application de la technique du Word2Vec dans la conception du dictionnaire, dans le domaine du terrorisme.

Ce projet nous a été très bénéfique, il nous a permis d'approfondir nos connaissances sur différentes notions, telles que le Web scraping et le traitement de texte, il nous également permis d'acquérir des connaissances sur le domaine de l'analyse des sentiments, ainsi que sur des techniques de Machine Learning. De plus, ce travail nous a aussi permis de nous familiariser avec l'utilisation de l'APIs Twitter, et des APIs en général. Et surtout, il nous a beaucoup appris sur comment effectuer et rédiger un travail de recherche.

BIBLIOGRAPHIE

- [1] <https://www.journaldunet.com> Consulté le 02/07/2020.
- [2] <https://www.blogdumoderateur.com/internet-reseaux-sociaux-2020> Consulté le 08/07/2020.
- [3] Concordiam. Extremism hits home stopping the spread of terrorism. J. Eur. Secur. Defense Issues, vol. 5, no. 3, p. 167, 2014.
- [4] Sentiment analysis of Twitter messages using Word2vec by weighted average. Kamel Ahcene Djaballah , Kamel Boukhalfa , Omar Boussaid, Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), 2019.
- [5] An approach for offensive text detection and prevention in Social Networks, S. H. Yadav and P. M. Manwatkar, in *ICIIECS-2015*.
- [6] Sentiment Analysis of Extremism in Social Media from Textual Information Muhammad Asif, Atiab Ishtiaq, Haseeb Ahmad, Hanan Aljuaid, Jalal Shah, Telematics and Informatics V. 48, May 2020.
- [7] <http://glossaire.infowebmaster.fr/web-2-0/> Consulté le 02/07/2020.
- [8] <https://www.definitions-marketing.com/definition/reseaux-sociaux/> Consulté le 29/07/2020.
- [9] <https://swello.com/fr/blog/linkedin-pourquoi-etre-present/> Consulté le 29/07/2020.
- [10] <https://www.futura-sciences.com/tech/definitions/informatique-twitter-10997/> Consulté le 29/07/2020.
- [11] <https://fr.malavida.com/faq/facebook/android/quest-ce-que-facebook-et-a-quoi-sert-il.html> Consulté le 29/07/2020.
- [12] <https://fr.vikidia.org/wiki/Sentiment> Consulté le 02/07/2020.
- [13] <https://fr.vikidia.org/wiki/Sentiment> Consulté le 02/07/2020.
- [14] <https://www.linternaute.fr/dictionnaire/fr/definition/opinion/> Consulté le 02/07/2020.
- [15] <https://fr.wikipedia.org/wiki/Opinion> Consulté le 02/07/2020.
- [16] Thibaut Thonet, Modèles thématiques pour la découverte non supervisée de points de vue sur le Web, l'université Toulouse 3 Paul Sabatier, 2017.
- [17] http://edutechwiki.unige.ch/fr/Analyse_de_sentiments_en_text_mining Consulté le 02/07/2020.
- [18] B. Liu, "Sentiment analysis and opinion mining", Synthesis lectures on human language technologies, vol. 5, pp. 1-167, 2012.
- [19]____Bing Liu, Opinions, Sentiment, and Emotion in Text, Cambridge University Press, 2015.
- [20] Antoine Cornuéjols, Cours d'I.A. "Introduction à la logique floue" 3° année, p.12, 2013

- [21] Christopher Johnson, Parul Shukla, Shilpa Shukla, "On Classifying the Political Sentiment of Tweets", University of Texas at Austin C Johnson, 2012.
- [22] Joshua Acosta, Norissa Lamaute, Mingxiao Luo, Ezra Finkelstein, and Andreea Cotoranu, Sentiment Analysis of Twitter Messages using Word2Vec, Seidenberg School of CSIS, Pace University, Pleasantville, New York 2017
- [23] Improving cyberbullying detection using Twitter users' psychological features and machine learning. Vimala Balakrishnana, Shahzaib Khana, Hamid R. Arabnia, computers & security volume 80, March 2020.
- [24] Faiza Belbachir, Expérimentation de fonctions pour la détection d'opinions dans les blogs, université de Paul Sabatier, Institut de Recherche en Informatique de Toulouse 2010.
- [25] <https://www.inbenta.com/fr/blog/analyse-de-sentiments-challenges-et-benefices/> consulté le 29/07/2020.
- [26] Pang B. et Lee L. Opinion Mining and sentiment analysis Foundations and Trends in Information Retrieval, 2008.
- [27] Vishal A. Kharde and S.S. Sonawane, TSentiment Analysis of Twitter Data: A Survey of Techniques, International Journal of Intelligent Systems and Applications (IJISA), 2016.
- [28] Kara Mostefa, Melouk Abdelbasset. Analyse des sentiments dans les réseaux sociaux en dialecte algérien Spécialité SDIA université d'el oued 2018-2019.
- [29] Alexander Pak and Patrick Paroubek, Twitter as a Corpus for Sentiment Analysis and Opinion Mining, Université de Paris-Sud, Laboratoire LIMSI NRS, France 2010.
- [30] Conde-Céspedes, P., Chavando, J., and Deberry, E., Detection of Suspicious Accounts on Twitter Using Word2Vec and Sentiment Analysis. In International Conference on Multimedia and Network Information System pp. 362-371. Springer, Cham. 2018.
- [31] Asif, M., Ishtiaq, A., Ahmad, H., Aljuaid, H., Shah, J., Sentiment Analysis of Extremism in Social Media from Textual Information, Telematics and Informatics (2020).
- [32] Sohail Ahmed, M. Qadoos., Terrorism Detection by Tweet Sentimental Analysis, August, 2018.
- [33] A. Rattrout and A. Ateeq., Sentiment Analysis of Social Networks Using Hybrid System: Dictionary-Based and Fuzzy Logic. July 1–2, 2019.
- [34] <https://www.linternaute.fr/dictionnaire/fr/definition/web-2-0/> Consulté le 10/07/2020.
- [35] Opinion mining and sentiment analysis openedition press Audrey Lohard et Dominique Boullier, 2012.
- [36] <https://webump.fr/blog/comment-fonctionne-instagram/> Consulté le 29/07/2020.
- [37] https://fr.wikipedia.org/wiki/Web_2.0 Consulté le 30/07/2020.
- [38] <https://machinelearningmastery.com/what-are-word-embeddings/> Consulté le 30/07/2020.

- [39] https://fr.wikipedia.org/wiki/Word_embedding Consulté le 30/07/2020.
- [40] <https://dataanalyticspost.com/Lexique/word2vec/> Consulté le 30/07/2020.
- [41] <https://www.guru99.com/word-embedding-word2vec> Consulté le 30/07/2020.
- [42] Hybrid Approach for Sentiment Analysis of Twitter Posts Using a Dictionary-based Approach and Fuzzy Logic Methods: Study Case on Cloud Service Providers Jamilah Rabeh Alharbi, King Abdul-Aziz, Wadee S. Alhalabi, University, Jeddah, Saudi Arabia, January-March 2020.
- [43] Porting Multilingual Subjectivity Resources across Languages. *Banea, Carmen, Mihalcea, Rada, Wiebe, Janyce.*
- [44] Finn Årup Nielsen A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. Proceedings of the ESWC 2011 Workshop on 'Making Sense of Microposts': Big things come in small packages 718 in CEUR Workshop Proceedings 93-98. Mai 2011.
- [45] The Lexicon-based Sentiment Analysis for Fan Page Ranking on Facebook. Phan Trong Ngoc, Myungsik Yoo. School of Electronic Engineering Soongsil University Seoul, Korea.
- [46] A new big data approach for topic classification and sentiment analysis of Twitter data. Rodrigues, Anisha P., Chiplunkar, Niranjana N.
- [47] Deep Recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. Al-Smadi, Mohammad, Qawasmeh, Omar, Al-Ayyoub, Mahmoud, Jararweh, Yaser, Gupta, Brij.
- [48] Subjectivity and Sentiment Analysis of Modern Standard Arabic Muhammad Abdul-Mageed, Mona T. Diab and Mohammed Korayem, 2011.
- [49] Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis, Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Proc. of HLT-EMNLP-2005.
- [50] [https://fr.wikipedia.org/wiki/Python_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage)) Consulté le 06 juillet 2020.
- [51] <https://fr.wikipedia.org/wiki/PyCharm> Consulté le 06 juillet 2020.
- [52] <https://fr.wikipedia.org/wiki/PyQt> Consulté le 06 juillet 2020.
- [53] Terrorism Detection Based on Sentiment Analysis Using Machine Learning Sofea Azrina Azizan and Izzatdin Abdul Aziz, 2017.
- [54] Fuzzy-Based Sentiment Analysis System for Analyzing Student Feedback and Satisfaction Yun Wang, Fazli Subhan, Shahabuddin Shamshirband, Muhammad Zubair Asghar, Ikram Ullah and Ammara Habib, CMC, vol.62, no.2, pp.631-655, 2020.
- [55] <https://medium.com/@mehdihadi/99ab87503a5e> Consulté le 04/08/2020.
- [56] <https://pypi.org/project/gensim/> Consulté le 04/08/2020.
- [57] <https://pypi.org/project/nltk/> Consulté le 04/08/2020.
- [58] <https://pypi.org/project/tweepy/> Consulté le 04/08/2020.

- [59] A. Tumasjan, T. Sprenger, P. Sandner, and I. Welp, "Predicting elections with twitter: What 140 characters reveal about political sentiment," in *Int. Conference on Web and Social Media*, 2010.
- [60] C. Edwards and L. Gribbon, "Pathways to violent extremism in the digital era," *The RUSI Journal*, vol. 158, no. 5, pp. 40–47, 2013.
- [61] A. Ward, "ISIS's Use of Social Media Still Poses a Threat to Stability in the Middle East and Africa," *Georgetown Security Studies Review*, Tech. Rep., 2018.
- [62] https://planspace.org/20150607-textblob_sentiment/ Consulté le 04/08/2020.