ROYAUME DU MAROC
Université Mohammed V - RABAT -
جامعة محمد الخامس – الرباط –
Faculté des sciences
كلية العلوم
CENTRE D'ÉTUDES DOCTORALES - SCIENCES ET TECHNOLOGIES

N° d'ordre : 4009

THÈSE

en vue de l'obtention du : **DOCTORAT**

Structure de Recherche: Intelligent Processing and Security of Systems
Discipline : Sciences et Technologies
Spécialité : Intelligence artificielle et science de données

Présentée et Soutenue le : *09/11/ 2024*

par :

Imane ENNEJJAI

Modèles de détection automatique des fausses nouvelles basés sur l'intelligence artificielle

Devant le JURY :

Fouzia OMARY	PES	Faculté des Sciences, Université Mohammed V, Rabat	Présidente
Mohammed YOUSSFI	PES	Ecole Normale Supérieure de l'Enseignement Technique, Université Hassan II, Mohammedia	Examinateur/Rapporteur
Hassan ECHOUKAIRI	MCH	Faculté des Sciences, Université Mohammed V, Rabat	Examinateur/Rapporteur
Younes CHIHAB	MCH	École Supérieure de Technologie, Université Ibn Tofail, Kénitra	Examinateur/Rapporteur
Mohammed OUMSIS	PES	École Supérieure de Technologie de Salé, Université Mohammed V, Rabat	Examinateur
Abderrahmane EZ-ZAHOUT	MCH	Faculté des Sciences, Université Mohammed V, Rabat	Examinateur
Karim EL BOUCHTI	MC	École Supérieure de Technologie de Salé, Université Mohammed V, Rabat	Invité
Soumia ZITI	PES	Faculté des Sciences, Université Mohammed V, Rabat	Directrice de thèse

Année Universitaire : 2023 - 24

« À mes chers parents,

Aucune dédicace ne saurait exprimer le respect, la gratitude éternelle et la profonde considération que j'ai pour vous deux. Papa, tu as consenti des sacrifices inestimables pour mon instruction, faisant bien plus que ce qu'un père pourrait imaginer, pour que nous suivions le bon chemin dans la vie et dans nos études. Maman, tu es pour moi le symbole de la bonté, la source de tendresse et l'exemple parfait de dévouement. Aucun mot ne pourrait véritablement exprimer l'amour que je ressens pour toi. Vos efforts conjoints, de jour comme de nuit, ont façonné mon éducation et assuré mon bien-être. Je vous dois toutes mes réussites et tout mon succès. Ce travail est dédié à vous, en témoignage de mon amour profond et sincère.

À mon mari,

À toi qui es mon pilier, mon soutien inébranlable et mon confident. Chaque pas dans ce voyage n'aurait été possible sans ta présence aimante à mes côtés. Ton amour, ta patience et ta foi en moi m'ont donné la force d'avancer, même dans les moments les plus difficiles. Je te remercie pour tout ce que tu fais, pour ton soutien inconditionnel et pour être toujours là, avec ton amour infini. Cette réussite est aussi la tienne.

À mon frère et à ma sœur, pour votre présence, votre complicité et vos paroles réconfortantes qui ont illuminé mon chemin.

À mon amie, pour ton soutien indéfectible, ta bienveillance et ton écoute précieuse tout au long de ce parcours.

À toute ma famille et mes proches,

Je tiens à exprimer toute ma gratitude pour votre amour, votre soutien inconditionnel et vos encouragements tout au long de ce voyage. Chacun d'entre vous a contribué à sa manière à cette réalisation, et je vous en suis profondément reconnaissante. Votre présence dans ma vie est une véritable bénédiction. »

Imane ENNEJJAI

Remerciements

Les recherches menées dans cette thèse ont été réalisées au sein de structure de recherche Traitement intelligent et sécurité des systèmes (IPSS) de la Faculté des Sciences de Rabat sous la direction du Madame **Soumia ZITI**.

Je tiens à exprimer mes sincères remerciements a Madame **Fouzia OMARY** professeure à la Faculté des Sciences de Rabat, de m'avoir intégré au sein de son équipe avec bienveillance et professionnalisme.

En premier lieu, je souhaite exprimer ma gratitude envers ma directrice de thèse, Madame **Soumia ZITI** professeure à la Faculté des Sciences de Rabat, pour son accompagnement, sa patience, sa grande disponibilité, ses qualités humaines et scientifiques, ainsi que pour toute l'assistance qu'il m'a apportée tout au long de la réalisation de ce travail.

J'exprime ma sincère reconnaissance envers a Madame **Fouzia OMARY** professeure à la Faculté des Sciences de Rabat, pour sa disponibilité et pour m'avoir fait l'honneur d'être président de ma soutenance.

Je tiens à exprimer ma profonde gratitude envers Monsieur **Mohammed YOUSSFI** professeur à l'école Normale Supérieure de l'Enseignement Technique, Université Hassan II deMohammedia. Je le remercie pour avoir accepté d'évaluer mon travail et de m'avoir honoré en tant que rapporteur.

Je tiens à exprimer ma gratitude à Monsieur **Hassan ECHOUKAIRI** maître de Conférence Habilité à la faculté des Sciences de Rabat. Je le remercie pour sa disponibilité et pour avoir accepté d'évaluer cette thèse en tant que rapporteur.

Je remercie également Monsieur **Younes CHIHAB** maître de Conférence Habilité à l'école Supérieure de Technologie, Université Ibn Tofail, Kénitra, pour avoir accepté d'évaluer mon travail et d'en être un rapporteur.

Je tiens à remercier chaleureusement Monsieur **Mohammed OUMSIS** professeur à l'école Supérieure de Technologie de Salé, Université Mohammed V, Rabat pour avoir examiné ce rapport avec rigueur et grand intérêt.

Je tiens à exprimer ma gratitude à Monsieur **Abderrahmane EZ-ZAHOUT** maître de Conférence Habilité à la faculté des Sciences de Rabat. Je le remercie pour sa disponibilité et pour avoir accepté d'évaluer cette thèse.

je souhaite exprimer ma reconnaissance à Monsieur **Karim ELBOUCHTI** maître de Conférence à la faculté des des Sciences Semlalia, Université kadi Ayyad, Marrakech, pour l'honneur qu'il m'a fait en acceptant de participer en tant qu'invité à cette soutenance. Sa présence témoigne de son intérêt pour ce travail, et je lui en suis profondément reconnaissant.

Enfin, je ne pourrai pas oublier tous ceux qui ont apporté leur contribution, de près ou de loin, à la réalisation de cette tâche.

Résumé

L'objectif de cette thèse est d'apporter des contributions significatives à la détection automatique des fausses nouvelles, en proposant un modèle innovant qui combine des techniques avancées de traitement du langage naturel et des architectures de réseaux neuronaux. Face à la diffusion rapide et massive de fausses informations sur Internet, ce travail s'attaque aux limites des approches existantes en termes de précision, d'efficacité et d'adaptabilité aux différents contextes. Après une définition générale et bien détaillé de fausses informations et de procédure de détections, une analyse comparative approfondie des méthodes actuelles de machine Learning et de deep learning sur plusieurs jeux de données en appliquant les techniques d'extraction de traitement du langage naturel, nous avons développé un modèle hybride combinant les réseaux neuronaux convolutifs, les réseaux à mémoire à long terme bidirectionnels et les réseaux attentionnels hiérarchiques. Ce modèle permet de capturer à la fois les dépendances locales et globales dans les données textuelles, tout en offrant une meilleure interprétabilité et robustesse face aux variations des jeux de données. Les résultats expérimentaux montrent que cette approche améliore significativement la précision de la détection des fausses nouvelles, ouvrant des perspectives pour des applications dans des domaines critiques tels que la finance, la santé, et la sécurité.

Mots-clefs : Détection des fausses nouvelles, Modèle hybride, Les réseaux neuronaux convolutifs, Les réseaux à mémoire à long terme bidirectionnels , Les réseaux attentionnels hiérarchiques. , Techniques NLP.

Abstract

The objective of this thesis is to make significant contributions to the automatic detection of fake news by proposing an innovative model that combines advanced natural language processing techniques with neural network architectures. In response to the rapid and widespread dissemination of false information on the Internet, this work addresses the limitations of existing approaches in terms of accuracy, efficiency, and adaptability to different contexts. After a general and detailed definition of fake news and detection procedures, as well as an in-depth comparative analysis of current machine learning and deep learning methods across several datasets using natural language processing extraction techniques, we developed a hybrid model combining convolutional neural networks, bidirectional long short-term memory networks, and hierarchical attention networks. This model captures both local and global dependencies in textual data, while offering better interpretability and robustness against variations in datasets. Experimental results show that this approach significantly improves the accuracy of fake news detection, opening up opportunities for applications in critical domains such as finance, healthcare, and security.

Keywords : Fake News Detection,Hybrid Model, Convolutional neural networks, Bidirectional long short-term memory networks , Hierarchical attention networks.

ملخص

هدف هذه الرسالة هو تقديم مساهمات كبيرة في كشف الأخبار الكاذبة تلقائيًا من خلال اقتراح نموذج مبتكر يجمع بين تقنيات متقدمة في معالجة اللغة الطبيعية وهندسيات شبكات العصب العميقة. تستجيب هذه العملية للانتشار السريع والواسع للمعلومات الزائفة على الإنترنت، وتتعامل مع القيود المتعلقة بالدقة والكفاءة والقابلية للتكيف مع سياقات متنوعة للنهجات الحالية. بعد تحديد شامل ومفصل للأخبار الكاذبة وإجراءات الكشف، بالإضافة إلى تحليل مقارن عميق للأساليب الحالية في التعلم الآلي والتعلم العميق عبر عدة مجموعات بيانات باستخدام تقنيات استخراج المعالم من معالجة اللغة الطبيعية، قمنا بتطوير نموذج هجين يجمع بين شبكات العصب العميقة التابعة للتعلم التلقائي، وشبكات الذاكرة القصيرة والطويلة ذات الاتجاهين، وشبكات الانتباه الهرمية. يتميز هذا النموذج بالقدرة على التقاط الاعتمادات المحلية والعالمية في البيانات النصية، مع توفير تفسير أفضل وصلابة أكبر ضد التغيرات في مجموعات البيانات. تظهر النتائج التجريبية أن هذا النهج يحسن بشكل كبير دقة كشف الأخبار الكاذبة، مما يفتح الفرص لتطبيقات في مجالات حيوية مثل الأمن والرعاية الصحية والمالية.

الكلمات الرئيسية : كشف الأخبار الكاذبة، نموذج هجين، شبكات العصب العميق التابعة للتعلم التلقائي، شبكات الذاكرة القصيرة والطويلة ذات الاتجاهين، شبكات الانتباه الهرمية.

Liste des Figures

Liste des Tables

Liste des Abréviations

FN	Fake news (Fausses informations)
NLP	Natural Language Processing (Traitement du Langage Naturel)
NN	Neural Network
CNN	Convolutional Neural Networks (Réseaux Neuronaux Convolutionnels)
BiLSTM	Bidirectional Long Short-Term Memory (Mémoire à Long Terme Bidirectionnelle)
HAN	Hierarchical Attention Networks (Réseaux Hiérarchiques d'Attention)
BOW	Bag of words (Sac de mots)
TF-IDF	Term Frequency-Inverse Document Frequency (Fréquence des Termes-Fréquence Inverse des Documents)
ML	Machine Learning (Apprentissage Automatique)
DL	Deep Learning (Apprentissage Profond)
RNN	Recurrent Neural Networks (Réseaux Neuronaux Récurrents)
SVM	Support Vector Machines (Machines à Vecteurs de Support)
F1 Score	F1 Score (Score F1)
AUC	Area Under the Curve (Aire Sous la Courbe)
BERT	Bidirectional Encoder Representation from Transformers
LSTM	Long-Short Memory
SGD	Descente de Gradient Stochastique
ReLU	Rectified Linear Unit
FPR	Le Taux de Faux Positifs
TPR	Le Taux de Vrais Positifs
NLTK	Natural Language Tool Kit

Table des Matières

Introduction

La diffusion de fausses informations constitue désormais l'un des enjeux majeurs de l'ère numérique. Avec l'essor des réseaux sociaux et des plateformes de partage en ligne, la désinformation se propage à une vitesse inégalée, influençant les opinions publiques, alimentant les divisions sociales et, dans certains cas, mettant en danger des vies humaines[9]. Qu'elles soient intentionnellement malveillantes ou simplement le résultat d'une mauvaise diffusion d'informations, ces fausses nouvelles menacent la confiance envers les médias, les institutions et les processus démocratiques. Chaque individu est aujourd'hui un créateur de contenu et peut publier des informations sans expertise ni responsabilité[10]. Ce phénomène a été exacerbé par des événements comme l'élection présidentielle américaine de 2016, la pandémie de COVID-19 ou les catastrophes naturelles telles que les tremblements de terre au Maroc en 2023, illustrant la rapidité avec laquelle les fausses informations se propagent.

L'Internet est devenu un vecteur rapide et économique pour la diffusion d'informations. Face à la masse d'informations disponible en ligne, il devient impossible pour un utilisateur ordinaire de vérifier systématiquement chaque donnée. Ce manque de connaissances spécialisées dans de nombreux domaines confère aux sources en ligne un crédit excessif. Cependant, en parallèle à la croissance des informations légitimes, la désinformation a explosé. Des informations fausses, souvent appelées "fake news", peuvent être largement diffusées jusqu'à devenir virales, motivées par des objectifs politiques, commerciaux ou même par ignorance. La pandémie de COVID-19 a illustré cette "infodémie"[11], où la désinformation sur la santé a mené à la méfiance envers les décideurs en matière de santé publique.

Les fausses nouvelles ne menacent pas seulement la démocratie et l'économie, mais peuvent également entraîner des décès. Par exemple, des incidents tragiques tels que "Pizzagate" ou les émeutes en Inde montrent que la désinformation peut avoir des conséquences mortelles. De plus, l'activité massive et la rapidité à laquelle les informations circulent sur les réseaux sociaux rendent leur analyse complexe et coûteuse[12]. Les utilisateurs souffrent de divers biais cognitifs, tels que l'effet de répétition, qui les rend plus susceptibles de croire à des informations fausses après plusieurs expositions, ou le biais de confirmation, qui les incite à accepter des informations correspondant à leurs croyances. Le contexte de cette recherche repose sur l'essor d'Internet comme vecteur rapide et économique pour la diffusion d'informations. Ce médium touche désormais un public bien plus large que n'importe quelle publication papier n'a jamais pu atteindre. Cependant, l'énorme quantité d'informations disponibles rend impossible une vérification systématique par les utilisateurs ordinaires, qui confèrent souvent un crédit excessif aux sources en ligne. Cela a contribué à l'explosion de la désinformation, aussi bien accidentelle que délibérée, créant un phénomène communément appelé "fake news"[13]. Ce phénomène est devenu un problème majeur auquel les utilisateurs du web sont confrontés, nécessitant des solutions automatisées

pour détecter, classifier, et supprimer les contenus malveillants.

Bien que de nombreuses solutions aient déjà été proposées, la conception de modèles de détection efficaces reste complexe, et l'explicabilité des modèles est souvent négligée au profit de la performance[14]. L'objectif de ce travail est de combler cette lacune en explorant plusieurs modèles d'apprentissage automatique populaires pour la détection des fausses nouvelles, en mettant l'accent sur l'explicabilité et l'influence de la représentation des données et de l'architecture des modèles sur leurs capacités à interpréter et classer les textes.

Malgré les progrès réalisés dans le domaine, une taxonomie contemporaine des modèles de représentation des caractéristiques et de classification n'a pas encore été développée. De plus, aucune étude comparative exhaustive n'a été menée en tenant compte des différentes techniques d'extraction de caractéristiques, des algorithmes de classification et des ensembles de données de référence. Cela soulève plusieurs questions de recherche : quel est l'impact des méthodes d'extraction de caractéristiques sur les performances des modèles ? La combinaison de ces méthodes améliorerait-elle la détection des fausses nouvelles ? Et quelles sont les méthodes les plus rentables ? Ce travail cherche à répondre à ces questions en menant une analyse approfondie des techniques disponibles et en identifiant des pistes de recherche pour développer des systèmes plus robustes. Ce travail cherche à combler ces lacunes en répondant à plusieurs questions de recherche fondamentales.

La premiére question de recherche concerne l'impact des méthodes d'extraction de caractéristiques sur les performances des modèles. Les méthodes d'extraction de caractéristiques, comme TF-IDF, les word embeddings (Word2Vec, GloVe)[15], ou encore les représentations basées sur des modèles de transformateurs (comme BERT), jouent un rôle crucial dans la qualité des données fournies aux algorithmes de classification. L'hypothèse initiale est que certaines méthodes d'extraction de caractéristiques peuvent mieux capturer les nuances sémantiques des fausses nouvelles que d'autres, ce qui se traduirait par des améliorations significatives des performances des modèles de classification. Pour répondre à cette question, une série d'expériences comparatives a été menée en utilisant plusieurs techniques d'extraction de caractéristiques sur différents ensembles de données. Les résultats préliminaires montrent que les techniques basées sur des modèles de transformateurs surpassent généralement les méthodes traditionnelles comme TF-IDF, en raison de leur capacité à capturer des relations contextuelles complexes dans les textes. Cependant, ces améliorations de performance sont souvent accompagnées d'une augmentation significative du coût computationnel, ce qui soulève des questions sur leur rentabilité dans des environnements contraints en ressources.

La deuxième question clé est de savoir si la combinaison de plusieurs méthodes d'extraction de caractéristiques pourrait améliorer les performances des modèles de détection des fausses nouvelles. L'idée sous-jacente est que différentes méthodes pourraient capturer des aspects complémentaires des données textuelles. Par exemple, alors que les modèles basés sur les transformateurs sont excellents pour saisir le contexte global d'un texte, des techniques comme TF-IDF ou les word embeddings peuvent mieux représenter des caractéristiques lexicales ou syntaxiques spécifiques.

Des expérimentations ont été réalisées pour tester diverses combinaisons de techniques d'extraction.

Les résultats montrent que l'intégration de méthodes multiples, permet d'obtenir des modèles plus robustes. Ces modèles combinés ont montré une amélioration moyenne en termes de précision et de rappel, par rapport à ceux utilisant une seule technique. Cette amélioration est due à une meilleure capture de la diversité des caractéristiques des fausses nouvelles, renforçant ainsi la capacité des modèles à différencier les informations fiables des fausses.

Cependant, bien que la combinaison de méthodes puisse augmenter les performances, elle accroît également la complexité du modèle et le coût en termes de temps de traitement et de ressources. Cette observation souligne l'importance de prendre en compte non seulement l'efficacité, mais aussi la rentabilité des approches dans des contextes d'application réels.

La rentabilité des méthodes, tant en termes de ressources computationnelles que de temps de traitement, est un aspect souvent négligé dans les études sur la détection des fausses nouvelles. Il est essentiel d'identifier les approches qui offrent le meilleur compromis entre performance et coût. À cet égard, les modèles basés sur des transformateurs, bien qu'étant les plus performants en termes de précision et de rappel, sont également les plus coûteux en termes de calcul. Leur complexité, combinée à la nécessité d'une puissance de calcul importante, limite leur utilisation dans des scénarios où les ressources sont limitées.

En revanche, les méthodes plus légères, comme les word embeddings traditionnels combinés avec des classificateurs simples offrent un excellent compromis. Ces modèles, bien que légèrement moins performants que les transformateurs, sont beaucoup plus rapides et moins gourmands en ressources, les rendant plus adaptés à des environnements où la rapidité et l'efficacité sont primordiales, par exemple pour une détection en temps réel des fausses nouvelles sur les réseaux sociaux. La détection des fausses nouvelles est devenue un enjeu critique pour les sociétés modernes, notamment à cause de ses impacts sociaux, politiques et économiques. Les statistiques montrent que la capacité humaine à distinguer les fausses nouvelles des vraies est comparable à un lancer de pièces, ce qui souligne l'importance de mettre en place des systèmes automatisés précis et fiables. La détection automatique des fausses nouvelles, bien que relativement récente, a suscité un intérêt croissant, particulièrement après l'élection présidentielle américaine de 2016[13], où l'impact des fausses nouvelles a été fortement ressenti.

Parmi les approches de détection des fausses nouvelles, les méthodes linguistiques basées sur les caractéristiques du contenu des nouvelles[16], telles que les textes des articles ou les publications sur les réseaux sociaux, se sont révélées particulièrement prometteuses. Cependant, peu d'études comparatives exhaustives ont été réalisées pour évaluer et comparer les différentes méthodes disponibles[17]. Certaines études ont exploré les modèles d'apprentissage automatique et d'apprentissage profond, mais n'ont pas analysé en profondeur les modèles de transformateurs ou les méthodes de bout en bout.

Cette thèse apporte une contribution significative au domaine de la détection des fausses informations en utilisant le traitement automatique du langage naturel et l'intelligence artificielle[18]. Nous proposons une approche innovante qui intègre une combinaison de trois modèles afin d'améliorer la précision et la fiabilité des systèmes de détection automatique des fausses nouvelles. Contrairement aux travaux précédents qui se concentraient sur des méthodes individuelles, cette recherche compare diverses techniques

d'extraction de caractéristiques et d'algorithmes de classification, y compris l'utilisation des extracteurs de caractéristiques de traitement de language naturel. En développant une taxonomie mise à jour et en évaluant la rentabilité des différentes approches, cette thèse ouvre de nouvelles perspectives pour la création de systèmes de détection plus robustes et efficaces.

Enfin, cette recherche propose plusieurs orientations futures, basées sur les conclusions de ce travail, visant à faire progresser le domaine de la détection des fausses informations. Ces contributions visent collectivement à améliorer l'efficacité et la fiabilité des systèmes automatisés pour identifier et atténuer la propagation des fausses informations.

Cette thèse est organisée en plusieurs chapitres, chacun traitant d'un aspect spécifique de la détection des fausses nouvelles.

— Chapitre 1 : Définition des Fausses Informations, Extraction de Caractéristiques et État de l'Art. Ce chapitre introduit les concepts clés autour des fausses informations, en détaillant les différentes formes de désinformation (désinformation intentionnelle et mésinformation). Il présente également les travaux connexes dans le domaine, ainsi que les méthodes d'extraction de caractéristiques basées sur le texte, le contenu visuel, le comportement des utilisateurs, et le contexte social. Enfin, les principales méthodes de détection, telles que la vérification des faits et les approches basées sur le contenu, sont explorées.

— Chapitre 2 : Processus de Détection des Fausses Informations Basé sur le Texte. Ce chapitre décrit les étapes du processus de détection de fausses nouvelles textuelles. Il couvre le prétraitement des données textuelles, la représentation des textes (méthodes de décompte et modèles prédictifs), et les algorithmes de classification utilisés. Il se concentre sur les modèles d'apprentissage automatique classiques et les réseaux neuronaux profonds, comme les CNN, BiLSTM, et HAN, ainsi que sur les fonctions d'activation et les optimiseurs.

— Chapitre 3 : Ensembles de Données et Métriques d'Évaluation. Ce chapitre présente les ensembles de données utilisés pour l'évaluation des modèles de détection des fausses nouvelles, tels que Liar Dataset et FakeNewsNet. Il explique également les principales métriques d'évaluation, comme la précision, le rappel et le score F1, nécessaires pour mesurer la performance des modèles.

— Chapitre 4 : Études Comparatives et Proposition de Contribution. Ce chapitre offre une analyse comparative des performances des différents modèles de détection de fausses nouvelles, incluant les approches basées sur CNN, BiLSTM et HAN. Il propose également une contribution sous forme d'un modèle combinant ces techniques pour améliorer la précision et la robustesse des systèmes de détection.

— Conclusion Générale et Travaux Futurs. La thèse se termine par une synthèse des résultats obtenus et propose des perspectives de recherche futures, notamment pour améliorer la robustesse des modèles et leur applicabilité dans de nouveaux domaines.

Chapitre 1

Fondements et État de l'Art en Détection

Introduction

Dans un contexte où l'information est diffusée à une vitesse sans précédent, la propagation des fausses nouvelles est devenue une problématique majeure. Leur diffusion rapide sur les réseaux sociaux et autres plateformes en ligne présente des risques importants pour la société, notamment en influençant les opinions publiques, en créant de la désinformation et en manipulant les comportements sociaux et politiques. Afin de comprendre pleinement ce phénomène et de proposer des solutions efficaces, il est essentiel d'examiner en profondeur les caractéristiques et les méthodes de détection des fausses nouvelles. Ce chapitre se propose de définir précisément ce que l'on entend par "fake news", d'explorer les travaux connexes dans ce domaine, et d'analyser les techniques d'extraction de caractéristiques. Enfin, nous examinerons les approches basées sur le contenu textuel ainsi que les méthodes actuelles pour détecter les fausses informations.

1.1 Definition du fausses informations

Les fausses informations ne sont ni quelque chose de nouveau ni quelque chose de trivial à résoudre, en fait, elles existent dans le monde depuis des siècles sous différents noms tels que la propagande ou les rumeurs. Nous avons examiné plusieurs définitions des fausses informations et combiné plusieurs de leurs attributs, aboutissant à une définition générique et à jour des fausses informations.

Géneralement une"fake news" peut être définie comme une information délibérément fausse ou trompeuse présentée comme vraie, souvent dans le but de manipuler l'opinion publique, de propager la désinformation ou d'atteindre des objectifs particuliers. Ce phénomène s'est intensifié avec l'avènement des médias sociaux et la facilité de diffusion rapide de l'information en ligne.

Selon [10, 13], il existe deux définitions reconnues des "Fake News" :

Définition 2.1 (Définition large) : Les "Fake News" sont des informations fausses.

Définition 2.2 (Définition étroite) : Les "Fake News" sont des articles de presse intentionnellement et vérifiablement faux.

À la différence de la première définition, qui se concentre exclusivement sur la validité de l'information, la seconde description souligne tant l'exactitude que les motivations sous-jacentes. Plus précisément, selon cette dernière, les fake news contiennent des informations trompeuses et sont destinées à induire en erreur les utilisateurs. De nombreux concepts liés aux fake news peuvent se recouper avec celui-ci. Ainsi, la définition étroite a été adoptée pour définir véritablement les fake news dans cette recherche, car elle permet d'éviter toute confusion avec d'autres concepts connexes pouvant être parfois utilisés à la place des fake news dans le processus de recherche.

Dans cette thèse, le terme Fake News est défini comme un texte qui est vérifiablement faux et diffusé avec une intention malveillante. Il y a trois aspects clés à cette définition. Premièrement, en se référant spécifiquement aux Fake News textuelles, d'autres sources médiatiques telles que la vidéo, les images ou l'audio sont exclues. S'attaquer à la détection des "deep fakes" nécessite des solutions IA différentes de celles travaillant avec du texte. En tant que deuxième aspect important, la définition implique que les Fake News peuvent être vérifiées. Il est donc possible de vérifier les affirmations énoncées comme vraies ou fausses. En incorporant cela dans la définition, les rumeurs sont exclues car il est souvent impossible de les vérifier. Les théories du complot relèvent de la catégorie des rumeurs car elles peuvent être considérées comme une rumeur à long terme qui énonce des affirmations difficiles à réfuter distinctement. Enfin, la définition cible l'intention des Fake News. Puisque cette intention doit être malveillante, toutes sortes de fausses nouvelles liées au divertissement telles que les canulars et les poissons d'avril sont exclus. De plus, l'intention est censée être malveillante dans le sens de vouloir influencer la discussion sociétale souvent en faveur d'une certaine propagande. Cela exclut également les textes publiés involontairement incorrects, par exemple avec des chiffres transposés.
Une illustration caractéristique de désinformation est la rumeur infondée propagée délibérément par des agents provocateurs russes concernant Hillary Clinton durant la course à la Maison Blanche de 2016, dans le but d'influencer l'opinion publique en faveur de Donald Trump. Dans ce cas, il est évident à quel point les Fake News peuvent être nocives. Mais il y a un problème supplémentaire lié au sujet des Fake News. Certaines Fake News sont simplement diffusées dans le but de susciter la méfiance des gens, de semer la confusion et d'empêcher les gens de pouvoir clairement distinguer le vrai du faux.

1.1.1 Types de fausses nouvelles

Les "fake news" désignent des informations publiées mais contenant des informations trompeuses afin de tromper intentionnellement les lecteurs [19] à des fins malveillantes [20]. La littérature montre qu'il existe plusieurs types de fausses nouvelles, , tel que le montre le schéma 1.1. Il s'agit de la rumeur, de la désinformation, de la mésinformation, du canular et de l'appât à clics [19]. Une rumeur est une déclaration non confirmée ou non étayée, qui se répand comme une traînée de poudre [21]. La désinformation est une information trompeuse délibérément publiée pour tromper les gens, tandis que la désinformation est une information inexacte partagée involontairement [19]. Lorsqu'un utilisateur publie de fausses informations avec une intention malveillante, il entre dans la catégorie de la désinformation

[22]. C'est le manque de connaissances des utilisateurs sur un sujet ou un domaine particulier qui est à l'origine de la diffusion de fausses informations [19]. Une catégorie de fausses nouvelles est un canular dont le but est d'induire intentionnellement le lecteur en erreur. Il s'agit notamment d'escroquer les utilisateurs et de leur faire perdre de l'argent [23]. Selon des études psychologiques, l'appât à clics est l'une des formes de fausses nouvelles qui attire les lecteurs en suscitant leur intérêt pour en savoir plus sur le titre accrocheur. Il les incite également à cliquer [35]. L'objectif de l'appât à clics est de rediriger les lecteurs vers de faux sites web afin d'augmenter le trafic vers des sites web contenant des publicités. Il s'agit d'un type de titre qui attire l'attention mais qui peut ne pas refléter le contenu de l'article [23].

La classification des divers formats de fausses informations est illustrée à l'aide d'un diagramme de Venn dans la Figure 1. Le Tableau 1.1 résume les différentes catégories ainsi que de l'impact du contenu frauduleux sur Internet.

TABLE 1.1 – Catégorisation de fausses informations

Catégorie	**Définition**	**impact**
Fausses nouvelles	Des informations fausses diffusées sous l'apparence de nouvelles authentiques, généralement propagées par le biais des médias d'information ou d'Internet, dans le but de gagner politiquement ou financièrement, d'augmenter le lectorat, ou d'influencer de manière biaisée l'opinion publique.	Pour nuire à une agence, une entité ou une personne ou pour tirer un profit financier/politique.
Rumeur	Une information non vérifiée qui n'est pas nécessairement fausse ; peut s'avérer vraie également.	Incertitude et confusion concernant les faits
Désinformation	Information délibérément trompeuse avec une intention prédéfinie	Promouvoir une croyance, une idée, un gain financier ou ternir l'image d'un adversaire
Clic appât	Utilisation délibérée de titres trompeurs pour encourager les visiteurs à cliquer sur une page web particulière	Générer des revenus publicitaires, déclencher des attaques de phishing
Canular	Récit faux, notamment par le biais d'une blague, d'une farce, de l'humour ou d'une tromperie malveillante, utilisé pour dissimuler la vérité	Le mensonge est perçu comme une vérité et une réalité

Satire/parodie	Articles contenant principalement de l'humour et de l'ironie, sans intention nocive mais ayant le potentiel de tromper. The Onion et Satire Wire sont des sources d'articles d'actualité satiriques.	"Le but est de s'amuser mais parfois peut avoir des effets néfastes
Spam d'opinion	Avis ou commentaires faux ou intentionnellement biaisés sur des produits et services"	"Opinion client mensongère
Propagande	Information injustement préjudiciable et trompeuse diffusée dans des communautés ciblées selon une stratégie prédéfinie pour promouvoir un point de vue particulier ou un agenda politique	Profit politique/financier
Conspiracy theories	une explication d'un événement qui invoque une conspiration par des acteurs sinistres et puissants, souvent motivée politiquement, reposant entièrement sur des préjugés ou des preuves insuffisantes	Extrêmement nocif pour les personnes

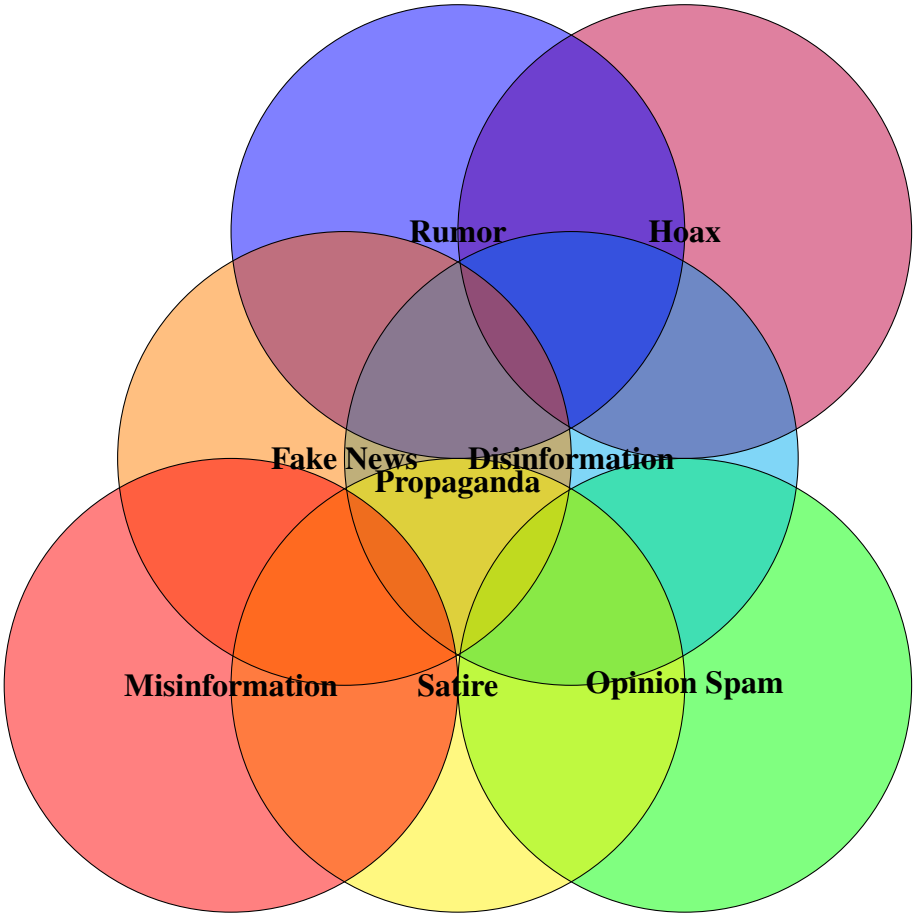


FIGURE 1.1 – Diagramme de Venn des fausses informations sur les médias sociaux et le web [1].

1.1.2 Désinformation

Depuis l'ère du conflit Est-Ouest(la guerre froide), le terme "désinformation" a été un sujet de discussion tant dans les médias que dans le milieu académique. Selon Martin [24], la désinformation est dérivée du mot soviétique « Dezinformatsiya », qui désigne la diffusion d'informations fausses et provocatrices. « Les services secrets de l'Union soviétique l'utilisaient comme une technique persuasive basée sur des falsifications et des événements mis en scène ». Il classait la falsification et la fabrication comme des parties intégrantes de la désinformation.

Bittman [25] affirme que le terme désinformation trouve ses racines dans l'allemand et a été adopté par les anciens Soviétiques pour tromper leurs adversaires idéologiques. Tant Martin que Bittman classifient la désinformation comme une forme de propagande.

— Des informations erronées, incorrectes ou trompeuses, élaborées, présentées et promues dans le but de causer un préjudice public ou de tirer profit de celles-ci.

— La désinformation fait référence à la création et à la diffusion délibérées d'informations que l'on sait être fausses (Wardle[26]).

— C'est une forme de propagande où soit le message repose sur un acte illégal, soit il déforme un acte légal ou une situation réelle. L'intention de la désinformation est de persuader par tous les moyens possibles ou disponibles (Martin [24]).

— La désinformation est détournée. Introduite secrètement dans le système de communication d'un adversaire, son intention est de tromper soit le public, auquel cas il s'agit de désinformation propagandiste, soit l'élite décisionnelle (Bittman[25]).

— La désinformation est une information intentionnellement fausse ou inexacte, délibérément diffusée [27].

— La désinformation n'est évidemment pas nouvelle. Documents falsifiés, photographies truquées, publicité trompeuse, cartes délibérément falsifiées et propagande gouvernementale [28].

— Désinformation : Des informations qui sont fausses et délibérément créées pour nuire à une personne, un groupe social, une organisation ou un pays [29].

Les sept définitions examinées et les termes connexes utilisés pour définir la désinformation incluent des concepts comme la fausseté, l'inexactitude, la tromperie, la création délibérée, une forme de propagande, la falsification, et la création intentionnelle. Les définitions semblent s'accorder sur l'intention de la désinformation : causer un préjudice public, obtenir un avantage financier, persuader, tromper et causer du tort au public ou à un pays.

1.1.3 Mésinformation

Plusieurs articles académiques définissent la mésinformation comme une erreur, une faute honnête ou une information inexacte. Elle n'a pas pour but de tromper, mais elle est trompeuse. (Fallis [28]). Wardle et Derakhshan [29] ont catégorisé trois types de désordre informationnel. En plus de la désinformation et de la mésinformation, ils incluent la mal-information, qui est définie comme une information basée sur la réalité, utilisée pour infliger du tort à une personne, une organisation ou un pays » [29].

— C'est une information erronée ou fausse circulée à la suite d'une erreur sincère, d'une omission, d'un préjugé ou simplement d'une ignorance (Bittman [25]).

— La mésinformation est la diffusion involontaire d'informations fausses [27].

— La mésinformation désigne le partage involontaire d'informations fausses [26].

— La mésinformation est une information inexacte résultant d'une erreur honnête ou de négligence [30].

— Des informations trompeuses ou inexactes partagées par des personnes qui ne les reconnaissent pas.

Les cinq définitions ci-dessus présentent une compréhension cohérente de la mésinformation. Il semble que le milieu académique ait une compréhension commune et moins ambiguë des définitions. Plusieurs articles et documents ont défini la mésinformation de la même manière, et nous les excluons pour éviter les répétitions.

1.2 Travaux connexes

Les fausses informations sont délibérément rédigées pour tromper le public et se composent de deux parties : l'authenticité et l'intention. L'authenticité signifie que les fausses informations contiennent des informations fausses qui peuvent être vérifiées comme telles, ce qui signifie que la théorie du complot n'est pas incluse dans les fausses informations car elle est difficile à prouver vraie ou fausse dans la plupart des cas. La deuxième partie, l'intention, signifie que les fausses informations ont été rédigées dans l'intention de tromper le lecteur. Les fausses informations existent depuis très longtemps, presque aussi longtemps que les nouvelles ont commencé à circuler largement après l'invention de l'imprimerie en 14397. Cependant, il n'existe pas de définition acceptée du terme "fausses nouvelles". Les fausses nouvelles sont rapidement devenues un problème sociétal, utilisées pour diffuser de fausses informations ou des rumeurs afin de modifier le comportement des gens. La diffusion de fausses informations a été démontrée comme ayant eu une influence significative sur les élections présidentielles américaines de 2016.

Des définitions plus larges des fausses nouvelles se concentrent sur l'authenticité ou l'intention du contenu des nouvelles. Certains journaux décrivent les nouvelles satiriques comme des fausses nouvelles car le contenu est faux, même si la satire est souvent divertissante et révèle sa propre tromperie aux consommateurs. D'autres publications qualifient directement les nouvelles trompeuses de fausses nouvelles, ce qui inclut des fabrications sérieuses, des canulars et des satires. Afin de développer la détection des fausses nouvelles, il est important de comprendre ce qu'elles sont et comment elles sont caractérisées. Ce qui suit est basé sur la perspective de l'exploitation minière de données des fausses nouvelles sur les médias sociaux [10].

L'étude réalisée par Shu, Silva, Wang, Jiliang, et Liu [10] offre une analyse approfondie des interactions des utilisateurs sur les médias sociaux et de leur importance dans la détection des fausses nouvelles. Les auteurs ont démontré que ces interactions jouent un rôle crucial dans la détermination de la véracité des informations, soulignant ainsi leur pertinence dans les modèles de détection. Leur recherche a introduit l'utilisation de caractéristiques linguistiques variées pour améliorer la précision des systèmes de détection de fausses nouvelles. Parmi ces caractéristiques, ils ont suggéré l'inclusion du nombre total de mots, de la longueur moyenne des mots, ainsi que des fréquences de grands mots et de phrases, utilisant des méthodes telles que les n-grammes et le sac de mots (bag of words). En outre, le balisage des parties du discours (POS) a été proposé comme une méthode complémentaire pour enrichir l'analyse linguistique. En explorant ces aspects, l'étude se concentre sur l'exploitation minière de données en tant qu'approche clé pour l'extraction de caractéristiques pertinentes. Les auteurs ont également abordé les différentes métriques d'évaluation utilisées pour mesurer l'efficacité des systèmes de détection et ont examiné des ensembles de données représentatifs, fournissant ainsi une vue d'ensemble complète des défis et des solutions dans le domaine de la détection de fausses nouvelles sur les médias sociaux.

L'étude menée par Wang et al. [31] a utilisé l'ensemble de données Liar pour comparer les performances de plusieurs modèles de détection des fausses nouvelles, y compris les SVM , la LR, les

Bi-LSTM et les CNN . Cette comparaison a permis de mettre en évidence les points forts et les limites de chaque approche. Les résultats ont montré que les réseaux neuronaux, en particulier les modèles basés sur les CNN, ont obtenu de bons résultats pour identifier les fausses nouvelles et suivre la propagation des informations par les utilisateurs. Wang a proposé un modèle hybride de réseau neuronal convolutif, qui s'est avéré surpasser d'autres algorithmes d'apprentissage automatique classiques en termes de précision et d'efficacité dans la détection des fausses nouvelles. Parallèlement, Shu et al. [10] ont exploré l'utilisation de trois sources d'informations auxiliaires disponibles sur les médias sociaux pour améliorer l'identification des nouvelles : les actualités elles-mêmes, les éditeurs de nouvelles, et les recruteurs de nouvelles. En établissant une connexion à trois voies entre ces acteurs, ils ont cherché à mieux cerner le contenu des nouvelles. Cependant, l'étude de Lina et al. [32] a révélé que leur modèle Bert, qui fonctionne à partir d'une seule direction d'inférence, pourrait avoir omis des informations importantes, ce qui pourrait limiter la précision de la détection. Ces recherches montrent l'évolution continue des techniques et des modèles pour affiner la détection des fausses nouvelles en tenant compte de diverses sources et méthodologies.

Ruchansky et al. [33] ont proposé un modèle de détection des fausses nouvelles innovant, baptisé CSI (Cross-Channel Social Influence), qui se distingue par son approche hybride profonde. Leur modèle intègre une variété de variables pour évaluer la véracité des nouvelles, en tenant compte notamment de l'engagement temporel entre un grand nombre d'utilisateurs et divers articles de presse au fil du temps. Cette approche vise à attribuer une étiquette de catégorisation des fausses nouvelles et à générer un score pour identifier les individus potentiellement suspects. Pour l'extraction des aspects temporels des articles de presse, les auteurs ont utilisé un RNN, tandis que les caractéristiques sociales ont été traitées à l'aide d'un réseau entièrement connecté. Les résultats des deux réseaux sont ensuite fusionnés pour réaliser la catégorisation finale des nouvelles. En ce qui concerne les caractéristiques textuelles, le modèle CSI a été testé sur deux ensembles de données distincts : l'un provenant de Twitter et l'autre de Weibo, la plateforme de microblogging chinoise. Selon les résultats, le modèle CSI a surpassé les modèles plus simples de 6 % en termes de précision, par rapport aux réseaux de base tels que le GRU (Gated Recurrent Unit). Ce résultat démontre l'efficacité de l'approche hybride de CSI pour améliorer la détection des fausses nouvelles en exploitant à la fois les dimensions temporelles et sociales des informations. Bajaj et al.[34] ont utilisé des réseaux neuronaux convolutifs pour aborder le problème d'un point de vue purement NLP (CNN). Le projet de l'Université Stanford vise à créer un classificateur qui peut déterminer si un matériau est vrai ou frauduleux uniquement sur la base de son contenu. Plusieurs architectures ont été étudiées, y compris une conception CNN novatrice qui inclut un mécanisme d'attention.

Dans leur étude, Tacchini et al. [35] ont exploré l'amélioration de la détection des fausses nouvelles en utilisant les caractéristiques des médias sociaux pour renforcer la fiabilité de leur modèle de classification. Leur approche se base sur deux techniques principales : la régression logistique et l'algorithme harmonique. La régression logistique est un modèle statistique largement utilisé pour la classification binaire, permettant de prédire la probabilité qu'une information appartienne à une catégorie spécifique, dans ce cas, celle des canulars ou des non-canulars. Cependant, Tacchini et al. ont introduit l'algorithme

harmonique comme une méthode complémentaire pour améliorer la performance de leur détecteur.

L'algorithme harmonique se distingue par sa capacité à transférer des informations entre les utilisateurs qui ont montré un intérêt commun pour certains messages. En d'autres termes, il utilise les interactions et les préférences des utilisateurs sur les médias sociaux pour relier et évaluer les messages en fonction de leur propagation et de leur réception au sein de différents groupes d'utilisateurs. Cette approche permet d'intégrer des informations contextuelles supplémentaires en tirant parti des comportements sociaux observés sur les plateformes. Les résultats de l'étude montrent que l'algorithme harmonique surpasse la régression logistique en termes de précision et d'efficacité pour la classification des informations, soulignant ainsi l'avantage de l'approche basée sur les interactions sociales pour détecter les canulars. En intégrant ces caractéristiques sociales, Tacchini et al.[35] ont réussi à améliorer significativement la performance de leur système de détection des fausses nouvelles.

Ahmed et al. [36] ont proposé une approche novatrice pour la détection automatique des fausses informations en se concentrant sur les avis et les nouvelles. Leur modèle repose sur l'utilisation d'n-grammes, une technique de traitement du langage naturel qui permet de capturer des séquences de mots contiguës dans le texte, afin d'améliorer l'identification des contenus trompeurs. Pour évaluer l'efficacité de leur modèle, les auteurs ont comparé deux méthodes d'extraction de caractéristiques : la fréquence des termes (tf) et la fréquence des termes inverse de la fréquence des documents (tf-idf). La première méthode mesure la fréquence absolue des mots dans un document, tandis que la deuxième ajuste cette fréquence en fonction de l'importance relative des mots dans l'ensemble des documents, permettant ainsi de mettre en avant les termes significatifs et réduisant l'impact des termes fréquents mais peu informatifs.

Ahmed et al. ont testé six algorithmes de classification d'apprentissage automatique pour déterminer lesquels sont les plus efficaces pour la détection des fausses informations. Parmi ces algorithmes, ils ont évalué à la fois des classificateurs linéaires et non linéaires. Les classificateurs linéaires tels que la Machine à Vecteurs de Support linéaire , la Descente de Gradient Stochastique et la Régression Logistique se sont révélés plus performants que les modèles non linéaires pour la classification des faux avis et des nouvelles. Cette observation suggère que les techniques linéaires, malgré leur simplicité apparente, peuvent offrir une meilleure précision dans la détection des informations trompeuses, probablement en raison de leur capacité à capturer efficacement les relations linéaires entre les caractéristiques textuelles.

Dans l'étude de Gupta et al. [37], les chercheurs ont développé un modèle semi-supervisé innovant pour la classification en temps réel des tweets, baptisé TweetCred. Ce modèle se distingue par son approche combinant à la fois des méthodes supervisées et non supervisées pour évaluer la crédibilité des tweets. Il a été formé en utilisant un vaste ensemble de données comprenant 5,4 millions de tweets, ce qui témoigne de la richesse et de la diversité des informations sur lesquelles le modèle s'appuie.

TweetCred intègre une variété de caractéristiques pour évaluer la véracité des tweets. Ces caractéristiques incluent :

— Les métadonnées : Informations complémentaires sur les tweets, telles que la date et l'heure de publication, qui peuvent fournir un contexte utile pour évaluer la crédibilité.

— Le contenu des tweets : Le texte même des tweets est analysé pour détecter des indices linguistiques ou des motifs qui pourraient indiquer la présence de fausses informations.

— Les informations linguistiques : Des caractéristiques telles que la structure du texte, le choix des mots et la syntaxe sont examinées pour identifier des anomalies ou des signes typiques de désinformation.

— L'auteur du tweet : Les attributs de l'utilisateur, comme son historique de publication, ses interactions et son influence, sont pris en compte pour évaluer sa crédibilité.

— Le réseau de l'auteur : L'analyse des relations entre l'auteur et ses abonnés, ainsi que les interactions dans le réseau social, peut fournir des informations supplémentaires sur la fiabilité de l'information partagée.

— Les liens inclus dans les tweets : Les URLs ou les liens partagés dans les tweets sont vérifiés pour déterminer leur authenticité et leur fiabilité.

En combinant ces diverses sources d'information, le modèle TweetCred a atteint une précision de plus de 80%, ce qui souligne son efficacité dans l'identification des tweets crédibles et non crédibles en temps réel. Cette performance est particulièrement remarquable compte tenu de la complexité et de la diversité des données traitées, ainsi que du défi inhérent à la détection des fausses informations dans un flux continu de contenu généré par les utilisateurs.

Dans l'étude menée par Volkova et al. [38], une approche innovante pour la vérification des tweets a été introduite, utilisant des réseaux neuronaux enrichis avec des données linguistiques. Cette méthode vise à améliorer la classification des tweets en différentes catégories, telles que l'authenticité ou la désinformation, en exploitant des caractéristiques linguistiques et sociales spécifiques.

L'étude s'appuie sur un vaste ensemble de données composé de 130 000 tweets, ce qui permet d'avoir une base solide pour l'entraînement et l'évaluation du modèle. L'approche de vérification multi-classes proposée repose sur plusieurs types de caractéristiques :

— Le texte des tweets : Les données textuelles sont analysées pour identifier les motifs et les indices susceptibles de signaler des biais ou de la subjectivité.

— Le graphe social : Les relations et les interactions au sein du réseau social sont prises en compte pour évaluer l'influence et la crédibilité des utilisateurs qui ont posté les tweets.

— Des marqueurs linguistiques de biais et de subjectivité : L'analyse linguistique approfondie aide à détecter les biais potentiels et la subjectivité dans le texte, des indicateurs cruciaux pour évaluer la fiabilité des informations.

— Des traits moraux : Les caractéristiques liées aux normes éthiques et morales sont examinées pour comprendre la nature des informations diffusées et leur impact potentiel sur les lecteurs.

Grâce à cette approche riche et multidimensionnelle, le modèle développé par Volkova et al. a atteint une précision remarquable de 95%. Ce résultat indique une capacité élevée à différencier entre les tweets crédibles et non crédibles, en intégrant non seulement les aspects textuels, mais aussi les dimensions

sociales et linguistiques. La combinaison de ces éléments permet au modèle de faire une évaluation plus complète et précise de la véracité des tweets, surmontant ainsi certains des défis typiques associés à la détection des fausses informations sur les réseaux sociaux.

Dans l'étude de Wei et al. [39], les auteurs ont développé un modèle non supervisé dédié à l'identification des titres trompeurs dans le domaine des nouvelles. Leur approche se distingue par l'absence de supervision explicite durant l'entraînement du modèle, ce qui signifie que le système ne nécessite pas de données étiquetées pour apprendre à détecter les titres trompeurs. Cette méthode permet d'explorer des aspects sous-jacents des titres sans être limité par les biais des étiquettes préexistantes.

Pour mener à bien cette tâche, Wei et al. [39] ont utilisé un ensemble de données substantiel composé de 40 000 nouvelles en chinois, ce qui assure une couverture étendue des variations possibles de titres trompeurs et authentiques. Les caractéristiques analysées par le modèle comprennent :

— Le nombre de mots : L'analyse du nombre de mots dans les titres permet de détecter des structures atypiques ou exagérées qui peuvent être associées à des titres trompeurs.

— Le nombre de chiffres : La présence et la fréquence des chiffres peuvent être des indicateurs de titres conçus pour attirer l'attention ou exagérer des informations.

— Le nombre de signes de ponctuation : Les titres contenant une ponctuation excessive ou atypique peuvent être suspectés de chercher à manipuler l'attention du lecteur.

— Les termes accrocheurs : L'utilisation de mots ou phrases accrocheurs est souvent un trait caractéristique des titres trompeurs, visant à susciter une réaction émotionnelle ou une curiosité non justifiée.

— Les sentiments : L'analyse des sentiments exprimés dans les titres permet d'évaluer s'ils sont excessivement positifs ou négatifs, ce qui peut être un indicateur de tromperie ou de manipulation émotionnelle.

— La distance entre les mots : L'examen de la distance entre les mots dans les titres peut révéler des structures linguistiques particulières associées aux titres trompeurs.

Le modèle proposé par Wei et al. [39] a démontré une précision de 65% dans l'identification des titres trompeurs. Bien que cette précision ne soit pas aussi élevée que certains modèles supervisés, elle représente un progrès significatif dans le développement de méthodes non supervisées pour la détection des titres trompeurs. Cette approche est particulièrement utile dans les contextes où des données étiquetées sont difficiles à obtenir, offrant une alternative viable pour l'analyse des titres de nouvelles et la détection de la tromperie.

Dans l'étude de Tabibian et al. [40], les chercheurs ont développé un cadre innovant pour modéliser les processus ponctuels temporels de réfutation et de vérification dans les référentiels de connaissances en ligne, en se concentrant particulièrement sur des plateformes collaboratives comme Wikipédia et Stack Overflow. Ce cadre vise à comprendre comment la crédibilité et la fiabilité des informations évoluent au fil du temps et en réponse à divers événements externes.

L'étude repose sur un vaste ensemble de données comprenant 19 millions d'événements extraits de 100 000 articles de Wikipédia, ainsi que 1 million d'événements provenant de Stack Overflow. L'ampleur de ces données permet une analyse approfondie des dynamiques de vérification et de réfutation des informations dans ces contextes en ligne.

Le modèle développé par Tabibian et al. examine plusieurs aspects critiques de la fiabilité des sources et des informations :

— Fiabilité des sources : Le cadre révèle que les sources les plus actives, c'est-à-dire celles qui contribuent fréquemment au contenu, tendent à être plus fiables. Cette observation s'explique par le fait que l'activité continue permet une meilleure surveillance et correction des erreurs. Toutefois, l'étude montre également que des sources moins actives peuvent également être crédibles, ce qui suggère que la fiabilité ne dépend pas uniquement de la fréquence des contributions mais aussi de la qualité et de la précision des informations fournies.

— Changements de fiabilité : Le modèle indique que les variations dans la fiabilité des articles sont souvent corrélées avec des événements externes notables, tels que des découvertes scientifiques, des changements politiques ou des crises sociales. Cette corrélation met en évidence comment les processus de vérification et de réfutation sont influencés par des événements pertinents et significatifs dans le monde réel.

— Questions et réponses sur Stack Overflow : L'analyse des questions et réponses sur Stack Overflow montre que ces éléments sont regroupés en fonction de leur popularité et de leur difficulté. Les questions populaires ou difficiles sont plus susceptibles d'attirer des réponses précises et bien vérifiées, ce qui contribue à l'amélioration générale de la qualité des informations sur la plateforme.

Le cadre proposé par Tabibian et al. offre une perspective précieuse sur les mécanismes de vérification et de réfutation des connaissances en ligne, soulignant l'importance de la dynamique temporelle et de la réponse aux événements externes dans l'évaluation de la crédibilité des informations. Cette approche enrichit notre compréhension des processus de validation des informations sur les plateformes collaboratives et peut guider le développement de systèmes plus efficaces pour la gestion de la qualité des contenus en ligne.

Ciampaglia et al.[41] ont proposé un modèle de vérification des faits en utilisant des graphes de connaissances et le calcul du plus court chemin entre des nœuds conceptuels basés sur des mesures de proximité sémantique. Le modèle utilise une base de données factuelle, DBpedia, contenant 3 millions de nœuds entités et 23 millions d'arêtes, et atteint une précision de plus de 70 %.

Castillo et al.[42] ont présenté un modèle de vérification de la crédibilité des déclarations sur les microblogs relatifs à des sujets tendance, basé sur l'apprentissage automatique. Le jeu de données comprenait 2 500 sujets différents et 10 000 tweets, avec des caractéristiques comme le comportement de tweet et de retweet, le contenu des déclarations et les liens vers des sources externes. Le modèle a obtenu une précision de plus de 70 %.

Kwon et al.[43] ont développé un modèle de détection des rumeurs en fonction des schémas de propagation cumulés des rumeurs dans le temps. Ce modèle s'appuie sur 1,7 milliard de tweets et utilise des caractéristiques basées sur l'utilisateur, la structure, le langage et le temps. Les résultats ont montré que les caractéristiques des utilisateurs sont prédictives au début de la circulation des rumeurs, tandis que les caractéristiques linguistiques restent des indicateurs puissants et stables.

Wu et al.[44] ont proposé la plateforme NICE pour évaluer la crédibilité de l'information sur les médias sociaux, en collectant une base de données de rumeurs vérifiées sur Sina Weibo. Avec un ensemble de données de 936 événements, leur modèle a atteint une précision de 88 % pour les rumeurs et 94 % pour les non-rumeurs.

Liu et al.[45] ont conçu un algorithme d'apprentissage semi-supervisé capable de sélectionner les instances les plus informatives, maximisant ainsi l'influence des étiquettes d'experts. Ils ont utilisé le jeu de données "20 News Group" et ont atteint une précision de 95 % avec seulement 23 % des instances validées par un expert.

Tschiatschek et al.[46] ont développé l'algorithme Detective, qui utilise l'activité de signalement des utilisateurs pour détecter les fausses informations. Ils ont utilisé un jeu de données Facebook comprenant 4 039 utilisateurs et 88 234 liens, et leur algorithme a montré des performances compétitives tout en restant robuste même lorsque la majorité des utilisateurs sont des adversaires.

Liu et al.[47] ont introduit un modèle de détection précoce de fausses informations sur les médias sociaux, classifiant les chemins de propagation des nouvelles via des réseaux neuronaux récurrents (RNN) et des réseaux convolutifs (CNN). Le modèle a été testé sur des données de Weibo et Twitter, atteignant une précision de 92 % sur Twitter et 85 % sur Weibo.

Ma et al.[48] ont proposé un modèle basé sur des RNN pour apprendre des représentations cachées capturant la variation des informations contextuelles des publications au fil du temps. Utilisant 5 millions d'affirmations, leur modèle surpasse les approches traditionnelles de détection de rumeurs et détecte les rumeurs plus rapidement et avec plus de précision que les techniques existantes.

Ruchansky et al.[33] ont proposé le modèle CSI (Capture, Score, Integrate), un modèle hybride pour la détection de fausses informations. Ce modèle s'appuie sur des données provenant de Weibo et Twitter, analysant le texte d'un article, les réponses des utilisateurs et les utilisateurs sources promouvant l'article. Le modèle a atteint 89 % de précision sur Twitter et 95 % sur Weibo, dépassant les modèles existants.

Singhania et al.[49] ont introduit un détecteur automatique utilisant un réseau d'attention hiérarchique à trois niveaux (3HAN). Testé sur 40 000 articles de sites de fausses et authentiques nouvelles, le modèle a utilisé des plongements de mots (GloVe) et des caractéristiques de contenu, obtenant une précision de 97 %.

Popat et al.[50] ont développé un modèle de réseau neuronal qui agrège des signaux externes comme le langage et la fiabilité des sources pour détecter les fausses informations, rendant les prédictions transparentes. Ils ont testé leur modèle sur Snopes, Politifact, et un ensemble de données de rumeurs Twitter, atteignant 57 % de précision sur SemEval, 78 % sur Snopes et 67 % sur Politifact.

Zagkotsis et al.[51] ont proposé un modèle TI-CNN qui intègre des informations textuelles et visuelles pour détecter les fausses informations. Basé sur un corpus de 20 015 nouvelles (11 941 fausses et 8 074 vraies), leur modèle a utilisé des caractéristiques linguistiques, de ponctuation, et d'analyse de sentiment, atteignant une précision de 92 %.

Zagkotsis et al.[52] ont utilisé des plongements de mots (GloVe, Fasttext, Word2Vec) pour classifier des articles à texte long et des déclarations à texte court. Sur le corpus Fake News Corpus, leur modèle a atteint une précision de 81 % avec des plongements de mots et l'apprentissage profond.

Nassif et al.[53] ont mené une étude sur l'utilisation de huit modèles d'incorporation contextuelle pour la détection de fausses informations en arabe. L'étude inclut deux ensembles de données d'actualités arabes, analysant des classificateurs basés sur des transformateurs.

Hegazi et al.[54] ont proposé une méthode d'extraction d'informations à partir de textes arabes, incluant des étapes comme le nettoyage et l'enrichissement. Ils ont collecté des données textuelles arabes depuis des serveurs de réseaux sociaux, créant un ensemble de données bien structuré et curaté.

Shishah et al.[55] ont introduit un cadre basé sur BERT pour l'extraction automatisée de caractéristiques d'entités, surpassant d'autres méthodes testées en termes de précision, rappel et score F1.

Sorour et al.[56] ont proposé un modèle hybride CNN-LSTM pour la détection des fausses informations en arabe, atteignant une précision de 81,6 % sur un ensemble de données collectant des titres d'actualités arabes.

Alzanin et al.[57] ont conçu un système de classification à cinq catégories pour les tweets arabes, utilisant des caractéristiques linguistiques et le contenu, surpassant les scores existants avec une approche RNN-GRU.

Kula et al.[58] ont proposé un modèle basé sur des réseaux DNN entraînés avec la bibliothèque Flair, atteignant une précision de 99,8 % sur des ensembles de données disponibles sur Kaggle.

Sastrawan et al.[59] ont utilisé une approche combinant des embeddings de mots pré-entraînés, CNN, LSTM et ResNet, réduisant les déséquilibres de classes grâce à une augmentation de données, et obtenant de meilleures performances avec l'architecture LSTM.

Ahmad et al.[60] ont utilisé un outil LIWC pour extraire des caractéristiques textuelles d'articles de fausses et vraies informations, atteignant de bonnes performances avec des apprenants en ensemble sur plusieurs ensembles de données disponibles sur Kaggle.

La Figure 1.2 présente une courbe illustrant la précision des différentes études sur la détection des fausses informations. Cette courbe permet de visualiser les performances des modèles et des approches proposées par chaque étude en termes de précision, mesurée en pourcentage.cette figure fournit un aperçu visuel précieux des performances relatives des différents modèles et algorithmes de détection des fausses informations. Elle met en évidence à la fois les succès significatifs dans le domaine et les défis persistants qui pourraient nécessiter des améliorations ou des recherches supplémentaires pour améliorer la précision globale des systèmes de détection de fausses informations.

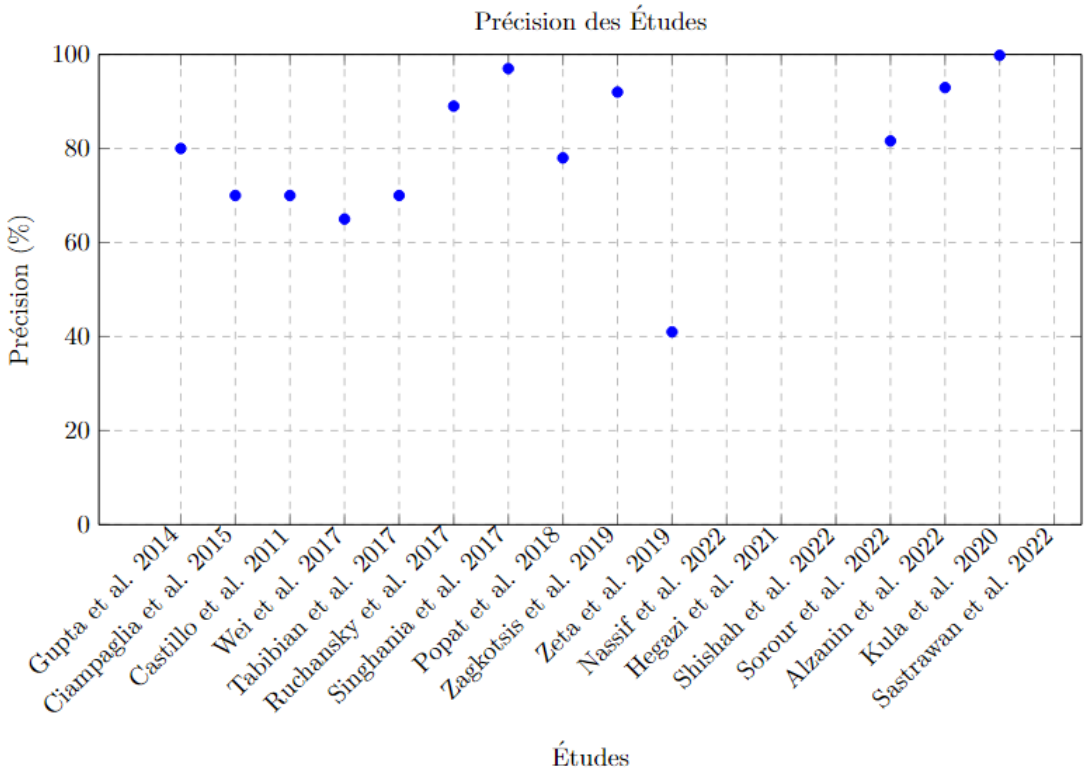


FIGURE 1.2 – Précision des Études sur la Détection des Fausses Informations

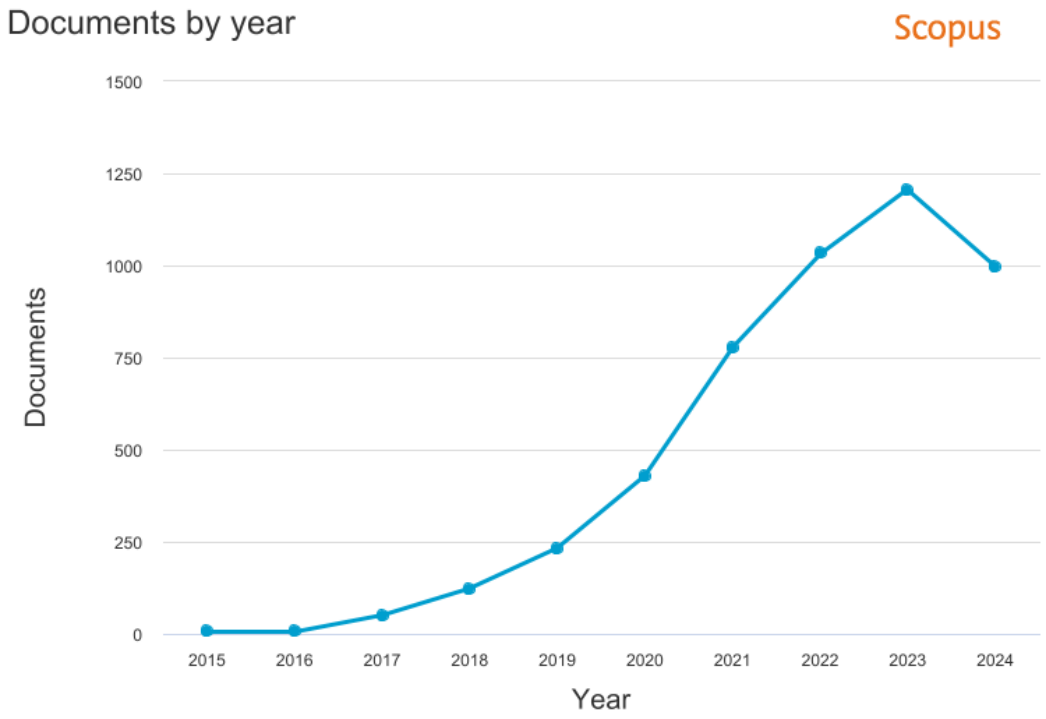


FIGURE 1.3 – Évolution du Nombre de Publications sur la Détection des Fausses Informations de 2014 à Aujourd'hui

La Figure 1.3 illustre l'évolution du nombre de publications sur la détection des fausses informations de 2014 à aujourd'hui. Cette figure montre une tendance continue à la hausse dans le volume des recherches publiées sur ce sujet au fil des années. L'axe des abscisses (x) représente les années, tandis que l'axe des ordonnées (y) indique le nombre de publications annuelles. On observe une croissance exponentielle depuis 2014, avec un nombre de publications qui ne cesse d'augmenter chaque année. Cette tendance souligne l'intérêt croissant pour la détection des fausses informations, en réponse à la montée en puissance des médias sociaux et à la prévalence des informations erronées dans l'espace numérique. En particulier, les années récentes affichent des pics notables dans la publication de nouveaux articles, ce qui reflète à la fois une intensification des efforts de recherche et une reconnaissance accrue de l'importance de ce domaine.

La figure révèle également que la recherche sur la détection des fausses informations est devenue un domaine de plus en plus dynamique et pertinent, avec une augmentation substantielle du nombre de contributions scientifiques. Cette tendance indique non seulement l'expansion continue des connaissances dans ce domaine mais aussi une nécessité croissante de solutions innovantes pour contrer la propagation des fausses nouvelles.la figure met en évidence la montée en puissance constante des publications sur la détection des fausses informations, témoignant d'une attention croissante et d'une intensification des efforts de recherche pour aborder ce problème complexe et omniprésent dans notre société numérique.

Les travaux connexes présentés révèlent une dynamique d'innovation et d'évolution dans le domaine de la détection des fausses informations, illustrant la variété des approches et des techniques employées pour faire face à ce défi. De l'approche non supervisée de Wei et al.[39] à l'utilisation de graphes de connaissances par Ciampaglia et al.[41], chaque étude apporte des contributions uniques qui enrichissent la compréhension des mécanismes sous-jacents à la tromperie médiatique.L'analyse des performances des modèles, comme le montre la Figure 2.2, indique des avancées significatives, mais également des limites qui nécessitent une attention particulière. Par exemple, bien que certains modèles atteignent des précisions élevées, la variabilité des résultats suggère que les systèmes de détection doivent encore être optimisés pour garantir une fiabilité accrue. Les caractéristiques examinées, telles que le nombre de mots, la présence de chiffres, et l'analyse des sentiments, démontrent l'importance d'une extraction de caractéristiques approfondie pour capturer les nuances du langage et des structures de titres trompeurs.La Figure 2.3 souligne la montée en puissance des publications sur la détection des fausses informations, indiquant un intérêt croissant de la communauté scientifique. Cela reflète non seulement la nécessité d'aborder la question des fausses nouvelles, mais aussi l'évolution des outils et des méthodologies disponibles pour analyser ce phénomène complexe.Cependant, malgré cette diversité d'approches, il est crucial de reconnaître que la détection des fausses informations ne se limite pas à l'analyse des images ou des vidéos, qui présentent leurs propres défis techniques. La phase de détection du texte revêt une importance majeure, car la majorité des fausses informations circulent sous forme écrite. Le texte, en tant que vecteur principal de la désinformation, nécessite une attention particulière, tant en ce qui concerne l'analyse linguistique que la compréhension des contextes sociaux et culturels dans lesquels ces informations sont diffusées.En

conséquence, cette recherche se concentrera exclusivement sur la phase de détection du texte. En affinant les techniques de traitement du langage naturel et en intégrant des modèles avancés tels que les réseaux neuronaux, nous visons à développer un système robuste capable d'identifier et de classer efficacement les informations trompeuses. Ce choix stratégique permettra de maximiser les ressources et les efforts de recherche pour faire face à un problème qui demeure d'une importance cruciale dans notre société numérique. La priorité accordée à la détection textuelle s'inscrit donc dans un contexte où la lutte contre la désinformation est plus pertinente que jamais, et où des solutions innovantes sont essentielles pour restaurer la confiance dans les médias et les informations disponibles au public.

1.3 Extraction de caractéristiques

Il existe plusieurs caractéristiques spécifiquement utilisées pour représenter les fausses nouvelles, qui sont divisées en trois catégories principales, à savoir les caractéristiques basées sur la création, les caractéristiques basées sur le contenu de l'actualité et les caractéristiques basées sur le contexte social.

1.3.1 Caractéristiques basées sur l'utilisateur

Les caractéristiques basées sur l'utilisateur(User-Based Features) sont conçues pour identifier des caractéristiques spécifiques des utilisateurs suspects ou des comptes non humains et sont classées en fonction du profilage des utilisateurs, de la crédibilité des utilisateurs et des caractéristiques du comportement des utilisateurs. Les caractéristiques de profilage des utilisateurs rassemblent des informations de base sur l'utilisateur, telles que leur nom de compte, leurs données de géolocalisation et leur date d'inscription. Les caractéristiques de crédibilité des utilisateurs évaluent l'impact et la fiabilité du compte en ligne, en tenant compte de caractéristiques telles que le score de crédibilité de l'utilisateur, le nombre d'amis et de followers, ainsi que le nombre total de publications ou de tweets. Les caractéristiques du comportement des utilisateurs sont destinées à identifier les schémas dans le comportement en ligne des utilisateurs légitimes et trompeurs. Ces caractéristiques font partie d'un ensemble plus large de caractéristiques de contexte social. Elles comprennent le score d'anomalie de l'utilisateur, qui est calculé en divisant le nombre d'interactions de l'utilisateur dans un laps de temps spécifique par sa moyenne mensuelle.

1.3.2 Caractéristiques basées sur le contenu

Les caractéristiques basées sur le contenu des actualités (News content-based features) peuvent également être utilisées pour détecter les fausses nouvelles. Ces caractéristiques peuvent être catégorisées en trois types : les caractéristiques basées sur la linguistique et la syntaxe, les caractéristiques basées sur le style, et les caractéristiques basées sur les aspects visuels. Ces caractéristiques fournissent des indices explicites pour la détection des fausses nouvelles et sont couramment utilisées pour la représentation et l'analyse des fausses nouvelles.

Les caractéristiques basées sur la linguistique et la syntaxe sont des aspects fondamentaux du langage naturel, incluant des composants tels que la structure et la sémantique. Ces caractéristiques sont précieuses pour analyser le contenu suspect des actualités, même si les fausses nouvelles sont intentionnellement générées pour induire en erreur les utilisateurs en ligne. Les caractéristiques basées sur la linguistique et la syntaxe peuvent être catégorisées en trois types :Les caractéristiques du mot, de la phrase et du contenu sont présentes. Les caractéristiques au niveau du mot incluent le sac de mots (bag-of-words), les n-grammes, la fréquence des termes (TF), le TF-IDF (Term Frequency-Inverted Document Frequency), etc. Les caractéristiques au niveau de la phrase incluent les parties du discours (POS), la longueur moyenne des phrases[61], la fréquence des ponctuations, etc. Enfin, les caractéristiques au niveau du contenu incluent les informations brutes sur le contenu méta des actualités[10].

Le but des caractéristiques basées sur le style est de distinguer les caractéristiques uniques des styles d'écriture entre les auteurs de fausses nouvelles et les auteurs de vraies nouvelles. Malgré les tentatives des auteurs de fausses nouvelles d'imiter le style d'écriture des auteurs de nouvelles légitimes, il existe encore des différences détectables qui peuvent être utilisées pour identifier les créateurs de fausses nouvelles.

Les caractéristiques basées sur les aspects visuels sont l'un des aspects les plus importants de la détection des fausses nouvelles, car les actualités comportant des éléments visuels ont tendance à être plus crédibles et à se propager plus rapidement. Les caractéristiques basées sur les aspects visuels, telles que le nombre d'images ou de vidéos, les scores de clarté et de cohérence, l'histogramme de distribution de similarité, le score de diversité, les scores de regroupement, et les ratios d'images, sont cruciales pour identifier les informations suspectes ou trompeuses dans le nouveau contenu.

1.3.3 Caractéristiques basées sur le contexte social

Les caractéristiques basées sur le contexte social (Social Context-based Features) sont destinées à montrer comment les actualités en ligne sont distribuées et comment les utilisateurs interagissent avec celles-ci. Ces caractéristiques peuvent être catégorisées en trois types : Les caractéristiques liées au réseau, les caractéristiques liées à la répartition et les caractéristiques liées au temps.

Analyse basée sur le réseau : L'analyse basée sur le réseau se concentre sur un groupe spécifique d'utilisateurs en ligne ayant des caractéristiques similaires telles que la localisation, le niveau d'éducation et les habitudes. Les caractéristiques basées sur le réseau sont sélectionnées et extraites de ces groupes pour étudier leurs caractéristiques uniques ainsi que les similitudes et les différences entre différents comptes en ligne.

Caractéristiques basées sur l'impact : Les caractéristiques basées sur l'impact sont utilisées pour capturer le schéma de diffusion unique des actualités en ligne, généralement en construisant un arbre de propagation pour décrire la nature de la distribution d'un article d'actualité. Les caractéristiques liées à l'arbre de propagation comprennent le degré de la racine, le nombre maximum de sous-arbres, et le degré et la profondeur maximum/moyenne de l'arbre.

Caractéristiques basées sur le temporel : Les caractéristiques basées sur le temporel sont utilisées pour décrire le comportement de publication des créateurs d'actualités en ligne dans une série temporelle, ce qui peut être utile pour détecter des activités de publication suspectes et indiquer le degré de fausseté des actualités en ligne. Les caractéristiques basées sur le temporel couramment utilisées comprennent l'intervalle entre deux publications, la fréquence de publication, de réponse et de commentaire pour un certain compte, l'heure de la journée à laquelle l'information originale est publiée/partagée/commentée, et le jour de la semaine où la publication est effectuée.

Dans cette thèse, nous travaillerons spécifiquement sur le contenu textuel et le contenu visuel. Par conséquent, nous donnerons une explication plus approfondie sur les caractéristiques textuelles et visuelles qui aident à la caractérisation des fausses nouvelles.

1.3.4 Le contenu visuel

La technologie multimédia fait référence à l'utilisation de plusieurs formes de médias telles que l'audio, la vidéo, les images et le texte pour transmettre des informations. Avec l'avancement rapide de la technologie multimédia, les actualités des médias personnels ont évolué des publications basées sur du texte vers des publications multimédias comprenant des images ou des vidéos. Cette évolution a permis de meilleures capacités de narration et a attiré davantage l'attention des lecteurs. Dans le passé, les actualités des médias personnels étaient principalement basées sur du texte. Bien que cette approche permettait des reportages et des analyses approfondis, elle manquait des éléments visuels qui peuvent donner vie à une histoire. Grâce à l'avancement de la technologie multimédia, les actualités des médias personnels peuvent désormais être enrichies avec des images, des vidéos et des sons qui aident les lecteurs à mieux comprendre et à se connecter au contenu.

Les publications multimédias sous forme d'images et de vidéos présentent plusieurs avantages par rapport aux publications basées sur du texte, car elles sont plus engageantes, captent plus rapidement l'attention et sont plus susceptibles d'être partagées sur les plateformes de médias sociaux. De plus, le contenu multimédia est également plus susceptible d'être mémorisé que le contenu textuel, ce qui en fait un outil puissant pour les créateurs d'actualités des médias personnels. De plus, l'utilisation de contenus multimédias dans les actualités des médias personnels a permis aux créateurs de raconter des histoires plus captivantes. En utilisant un mélange de différents formats de médias, les créateurs peuvent créer une expérience plus immersive qui aide les lecteurs à se connecter au contenu à un niveau plus profond. Les fausses nouvelles tirent parti de la technologie multimédia en utilisant des images falsifiées ou modifiées pour attirer et tromper les lecteurs, ce qui fait du contenu visuel une partie essentielle des fausses nouvelles qu'il ne faut pas ignorer.

1.4 Le contenu basée sur le texte

Dans cette méthode, nous cherchons des motifs à travers les caractéristiques textuelles qui sont extraites des articles d'actualité. Les caractéristiques textuelles font référence aux éléments tirés du titre ou du texte principal des articles d'actualité [10]. Il existe trois types de caractéristiques textuelles : les caractéristiques linguistiques, les caractéristiques textuelles à faible rang, et les caractéristiques textuelles neuronales. Les caractéristiques linguistiques analysent le texte à différents niveaux tels que les mots, les phrases et les syntagmes, et fournissent différentes caractéristiques des fausses nouvelles par rapport aux nouvelles factuelles [10]. Des exemples de caractéristiques linguistiques incluent les caractéristiques lexicales, syntaxiques et sémantiques (par exemple, TF, TF-IDF) [10, 62]. Les modèles à faible rang utilisent la factorisation de tenseurs ou de matrices pour extraire une représentation textuelle à petite échelle à partir d'une matrice de caractéristiques d'entrée à grande échelle et bruyante [10]. Les caractéristiques textuelles neuronales reposent sur des représentations vectorielles denses telles que les méthodes d'incorporation (Word2Vec, GloVe, fastText, etc.), plutôt que sur des caractéristiques à haute dimension et éparses [62]. Par conséquent, nous pouvons considérer la détection de fausses nouvelles basée sur le texte comme un problème de classification.

Malgré la plupart des études utilisant des modèles d'apprentissage supervisé pour détecter les fausses nouvelles, certaines études ont utilisé des modèles d'apprentissage non supervisé [63, 64], semi-supervisé [65, 11, 66] et par renforcement [67]. Dans cette étude, nous nous concentrons sur les études qui ont utilisé des caractéristiques linguistiques et des caractéristiques textuelles neuronales pour la détection de fausses nouvelles. Non seulement parce qu'elles ont montré la capacité de discriminer avec précision entre les vraies et les fausses nouvelles, mais aussi parce qu'elles sont couramment utilisées en raison de l'abondance de l'information textuelle présente dans toutes les actualités. De plus, il existe un manque de jeux de données avec d'autres types de caractéristiques.

Il existe une multitude d'études sur la détection de fausses nouvelles basée sur le texte. Nous pouvons classifier les travaux existants en fonction des définitions, des termes connexes, de la langue, des approches de détection de fausses nouvelles, et des méthodes d'extraction de caractéristiques. À cet égard, La langue la plus étudiée dans dans le contexte de la détection de fausses nouvelles est l'anglais. Cependant, des progrès récents ont également été réalisés dans de nombreuses autres langues comme le slovaque [68, 69], l'ourdou [70, 71, 72],l'indien [73], le chinois [74].... Il existe également des études qui associent les fausses nouvelles à des termes comme rumeur [75, 76], actualité satirique[77]....

Selon la tâche de détection de fausses nouvelles, la plupart des travaux utilisent des approches d'apprentissage supervisé. À cet égard, d'une part, certains travaux utilisent des classificateurs monolithiques pour classer les actualités. D'autre part, certains travaux utilisent des classificateurs par ensemble pour les tâches de classification des actualités. La plupart des articles récents utilisent davantage de méthodes d'incorporation de mots et de méthodes pré-entraînées que des méthodes Bag-of-Words, N-grammes et Vectoriseur de Comptage, car les méthodes d'incorporation de mots peuvent fournir des informations sur la relation entre les mots et représentent, par conséquent, une amélioration des performances.

En ce qui concerne les modèles de classificateurs, différents paradigmes ont été mis en œuvre ; des algorithmes basés sur les arbres tels que les arbres de décision (DT) et les forêts aléatoires (RF) [78, 79, 80, 81], des réseaux neuronaux artificiels tels que le perceptron multicouche (MLP) [82, 83] et les réseaux neuronaux convolutifs (CNN) [31, 84, 85], le bayésien naïf (NB)[78, 80], et les machines à vecteurs de support (SVM) [78, 79, 80, 81]. Les approches par ensemble peuvent comprendre des modèles d'ensemble profonds et des modèles d'ensemble d'apprentissage machine. La plupart des travaux utilisent des caractéristiques linguistiques pour entraîner les modèles ; cependant, certains travaux utilisent des caractéristiques visuelles [86, 12] ou des caractéristiques sociales [87, 88]. Il existe des études qui ont utilisé une approche d'apprentissage multi-vue [89, 90, 91].

1.5 Détection des fausses informations

Dans cette partie, nous décrirons les mécanismes de détection des fausses informations en nous concentrant sur les méthodes basées sur le texte.

1.5.1 Base de connaissances

Tester la véracité des déclarations principales dans un article de presse est le meilleur moyen d'évaluer la vérité des informations. Pour ce faire, des approches basées sur la connaissance, qui impliquent l'utilisation de sources externes pour vérifier les déclarations rapportées dans le journalisme, peuvent être utilisées. La vérification des faits est une méthode initialement développée dans le journalisme, qui vise à tester la validité des informations en confrontant les informations issues des articles de presse à vérifier (par exemple, ses revendications ou déclarations) avec des faits connus (par exemple, des informations vérifiées). Il y a eu un intérêt croissant pour la vérification des faits, et plusieurs tentatives ont été faites pour développer des programmes de vérification des faits automatisés et pratiques. La vérification des faits peut être catégorisée en trois types :

Vérification des faits basée sur des experts

La vérification des faits basée sur des experts est une méthode largement utilisée pour vérifier l'exactitude des reportages d'actualités. Cette approche implique l'utilisation d'experts du domaine ou de vérificateurs de faits pour évaluer la véracité des revendications de nouvelles. Ces experts sont généralement des individus ayant des connaissances spécialisées dans un domaine particulier, capables d'évaluer la validité des affirmations faites dans les articles de presse en fonction de leur expertise. La vérification des faits basée sur des experts est souvent effectuée par un petit groupe de vérificateurs de faits hautement fiables. Cela rend le processus de vérification des faits plus facile à gérer et peut conduire à des résultats raisonnablement précis. Cependant, cette approche peut être coûteuse car elle nécessite l'embauche et la formation d'un groupe d'experts du domaine pour mener le processus de vérification des

faits. Par conséquent, la vérification des faits basée sur des experts peut ne pas être scalable avec l'augmentation du volume de contenu d'actualités à examiner. À mesure que le volume d'articles de presse et de rapports continue d'augmenter, il peut devenir plus difficile d'embaucher suffisamment d'experts du domaine pour gérer la charge de travail. Cependant, bien que la vérification des faits basée sur des experts présente certaines limites, elle reste une approche importante et précieuse pour vérifier l'exactitude des reportages d'actualités. En exploitant l'expertise des experts du domaine, il est possible d'obtenir une compréhension plus approfondie des problèmes complexes et d'identifier les affirmations fausses ou trompeuses dans les articles de presse.

Vérification des faits basée sur la foule

La vérification des faits basée sur la foule est une technique pour déterminer l'exactitude du contenu d'actualités en utilisant un grand groupe de personnes ordinaires comme vérificateurs de faits. Contrairement à la vérification des faits basée sur des experts, qui est généralement effectuée par un petit groupe d'experts, la vérification des faits basée sur la foule repose sur l'intelligence collective de la foule. Cependant, l'approche basée sur la foule est moins fiable et précise en raison des biais politiques des vérificateurs de faits et des annotations contradictoires. Néanmoins, cette méthode peut être mise à l'échelle pour gérer un grand volume de contenu d'actualités. Le fonctionnement consiste à encourager les personnes ordinaires à annoter le contenu des actualités, puis à agréger ces annotations pour créer une évaluation globale de la véracité des actualités. Les sites Web de vérification des faits basés sur la foule en sont encore à leurs débuts, tout comme la vérification des faits basée sur des experts. Un exemple de site Web de vérification des faits basé sur la foule est www.fiskkit.com, qui permet aux utilisateurs de télécharger des articles et de fournir des notes pour chaque phrase de ces articles, ainsi que de sélectionner des balises pour décrire le contenu. Les sources citées dans les articles aident à différencier les contenus d'actualités des contenus non journalistiques et à évaluer leur crédibilité. Les balises catégorisées selon plusieurs dimensions permettent l'analyse des tendances à travers les articles de fausses et réelles nouvelles.

Vérification des faits orientée vers l'ordinateur

La vérification des faits orientée vers l'ordinateur est une méthode pour vérifier l'exactitude des reportages d'actualités qui utilise l'automatisation pour détecter les déclarations vraies et fausses. Elle aborde deux problèmes majeurs, à savoir la reconnaissance des déclarations crédibles et la distinction de la véracité des déclarations de fait. Cette méthode extrait des arguments factuels du matériel d'actualités pour faciliter le processus de vérification des faits. Pour évaluer la véracité d'un argument particulier, la vérification des faits orientée vers l'ordinateur s'appuie sur des ressources externes, telles que des sources web ouvertes, pour contraster avec les déclarations particulières analysées.

1.5.2 Basé sur le style

Les éditeurs de fausses informations ont souvent l'intention de répandre des informations trompeuses et fausses afin de manipuler un grand nombre de personnes. Pour atteindre cet objectif, ils utilisent différents types de stratégies d'écriture pour convaincre leur public. Ces stratégies diffèrent de celles utilisées par les éditeurs de vraies informations car elles visent à tromper les lecteurs plutôt qu'à fournir des informations factuelles et impartiales. Une manière d'identifier les fausses informations est à travers les stratégies basées sur le style qui se concentrent sur le format d'écriture. Cette approche implique d'analyser le style d'écriture et d'identifier les schémas couramment utilisés par les éditeurs de fausses informations pour manipuler leurs lecteurs. Il existe deux principales catégories d'approches basées sur le style : celles orientées vers la tromperie et celles orientées vers l'objectivité.

Les méthodes de stylométrie orientées vers la tromperie utilisent des techniques médico-légales pour identifier les déclarations trompeuses dans les articles de presse. Ces méthodes s'inspirent de la psychologie médico-légale et incluent l'analyse de contenu basée sur des critères et l'analyse de contenu basée sur des méthodes scientifiques. Des modèles plus avancés, tels que ceux basés sur l'analyse du langage naturel, sont également utilisés pour identifier différentes phases de tromperie, y compris la syntaxe approfondie et la structure rhétorique.

Les approches orientées vers l'objectivité visent à identifier les indices stylistiques suggérant un manque d'objectivité dans le contenu des actualités. Cela pourrait indiquer un potentiel de manipulation du public à travers des types hyper-partisans et du journalisme jaune. Les types hyper-partisans se réfèrent à des actions qui ciblent de manière disproportionnée un seul groupe politique, souvent dans le but de produire de fausses informations. Les caractéristiques basées sur la linguistique peuvent être utilisées pour identifier les articles hyper-partisans, tandis que le journalisme jaune se réfère à des articles qui s'appuient sur des titres accrocheurs et ont tendance à exagérer, à sensationnaliser ou à faire de la peur plutôt que de présenter des informations bien recherchées. Les titres trompeurs accrocheurs peuvent servir d'indicateur fiable d'articles inexactes et trompeurs, car les titres résument souvent les principaux points de vue que l'auteur souhaite exprimer.

1.5.3 Basé sur le contexte social

La nature des médias sociaux offre aux chercheurs des ressources supplémentaires pour améliorer les modèles de contenu d'actualités. Les modèles de contexte social impliquent la collecte d'engagements sociaux spécifiques des utilisateurs selon différentes perspectives pour compléter l'étude. Il existe deux catégories de méthodes de modélisation de contexte social : basées sur la position et basées sur la propagation. Les méthodes basées sur la position se concentrent sur l'identification de l'attitude ou de l'opinion des utilisateurs à l'égard d'un sujet particulier, tandis que les méthodes basées sur la propagation se concentrent sur l'analyse de la manière dont l'information se propage à travers les réseaux sociaux.

1.5.4 Contenu basé sur les éléments visuels

Les caractéristiques visuelles extraites des éléments visuels sont efficaces pour détecter les fausses informations. Cependant, des études limitées ont vérifié la valeur du contenu multimédia sur les médias sociaux. Récemment, diverses caractéristiques visuelles et statistiques ont été extraites pour la prédiction des actualités, y compris l'utilisation d'un cadre de classification pour reconnaître les fausses images basé sur des caractéristiques au niveau de l'utilisateur et au niveau du tweet. Une autre recherche a examiné plus en profondeur l'efficacité de plusieurs détecteurs d'images fausses utilisant des GAN (réseaux antagonistes génératifs) pour la conversion d'image à image, mais ces modèles sont toujours conçus à la main et complexes pour représenter le contenu visuel.

1.5.5 Contenu basé sur le texte

Les caractéristiques textuelles font référence aux caractéristiques statistiques ou sémantiques qui peuvent être extraites du contenu textuel des messages, telles que la fréquence des mots, la structure des phrases, l'analyse des sentiments et d'autres caractéristiques linguistiques. Ces caractéristiques sont souvent utilisées dans l'identification des fausses informations car elles peuvent fournir un aperçu du style d'écriture, du ton et du sentiment émotionnel du contenu. La plupart des recherches précédentes sur la détection des fausses informations ont largement reposé sur ces caractéristiques textuelles, ainsi que sur les métadonnées utilisateur, pour identifier et distinguer les fausses informations du contenu authentique. En analysant ces caractéristiques, il est possible d'identifier des motifs spécifiques et des points communs que l'on trouve couramment dans le contenu des fausses informations, permettant une détection plus précise et fiable.

De plus, les processus basés sur le traitement du langage naturel (NLP) impliquent une gamme de tâches, y compris le prétraitement, l'incorporation de mots et les techniques d'extraction de caractéristiques. Dans le contexte de la détection des fausses informations, plusieurs modèles utilisent le prétraitement des données comme étape initiale pour représenter les attributs obscurs, gérer les mots perdus, binariser les attributs et traiter les structures compliquées. Le prétraitement des données peut également aider à visualiser les données et à résoudre les problèmes de données bruyantes, tout en économisant de l'espace et du temps de calcul. La vectorisation de mots est un autre processus clé dans les modèles de détection des fausses informations basés sur le NLP. Elle consiste à mapper du texte ou des mots sur une liste de vecteurs, qui peuvent ensuite être utilisés pour l'analyse et la classification. Les techniques courantes de vectorisation de mots incluent le modèle de sac de mots et TF-IDF (fréquence de terme-fréquence inverse du document). Cependant, ces dernières années, les modèles d'incorporation de mots pré-entraînés tels que word2vec et GloVe sont devenus de plus en plus prisés grâce à leur capacité à traiter de grands ensembles de données et à améliorer la précision.

Conclusion

En conclusion, ce chapitre a permis d'établir une compréhension claire des fausses nouvelles, de leurs caractéristiques, et des différents défis associés à leur détection. Nous avons tout d'abord défini les fake news en soulignant la complexité de leur nature et leur impact sur la société. Ensuite, un examen des travaux connexes a permis de mettre en lumière les diverses approches existantes, tant au niveau des techniques d'extraction de caractéristiques que des méthodes basées sur le contenu textuel. La détection des fausses nouvelles nécessite une combinaison d'approches sophistiquées, telles que l'utilisation de techniques avancées en traitement du langage naturel et l'exploitation des caractéristiques textuelles. Les résultats des recherches actuelles montrent qu'il est possible de mieux comprendre et de détecter ces contenus nuisibles, bien que de nombreux défis restent à surmonter pour améliorer la précision, la performance et l'efficacité des méthodes proposées.

Chapitre 2

Processus de Détection de Fausses Informations Textuelles

Introduction

La propagation des fausses informations sur les plateformes numériques est devenue une problématique majeure dans notre société contemporaine. Ces informations erronées, souvent diffusées avec des intentions malveillantes, peuvent causer des dommages significatifs,à la fois au niveau personnel et au niveau social. La détection automatique de fausses informations est donc essentielle pour maintenir l'intégrité des informations circulant sur internet.

Ce chapitre explore en profondeur le processus de détection de fausses informations textuelles, en commençant par le prétraitement des données textuelles, en passant par les différentes méthodes de représentation textuelle, jusqu'à l'application de divers algorithmes de classification.

2.1 Processus de détection de fausses informations basé sur le texte

Dans cette thése nous limitons notre focus à la détection de fausses informations basée sur le texte 2.1, principalement parce que c'est la méthode la plus populaire utilisée par les chercheurs [10], mais aussi parce que les ensembles de données de référence pour la détection de fausses informations incluent généralement des caractéristiques du contenu d'actualités, avec un accent particulier sur le texte [92]. En considérant l'existence de jeux de données textuels étiquetés dans la détection automatique de fausses informations, cette tâche peut être formalisée comme une tâche de classification de texte (2.1). Supposons que x se réfère à un article de presse, nous avons donc besoin d'une fonction pour identifier s'il s'agit de fausses informations ou de vraies informations. Par conséquent, nous pouvons considérer H comme se référant à la fonction de prédiction. Ainsi, la tâche de détection de fausses informations peut être identifiée comme une fonction indicatrice :

$$H(x) = \begin{cases} 1 & \text{si } x \text{ est une fausse information} \\ 0 & \text{si } x \text{ est une vraie information} \end{cases} \tag{2.1}$$

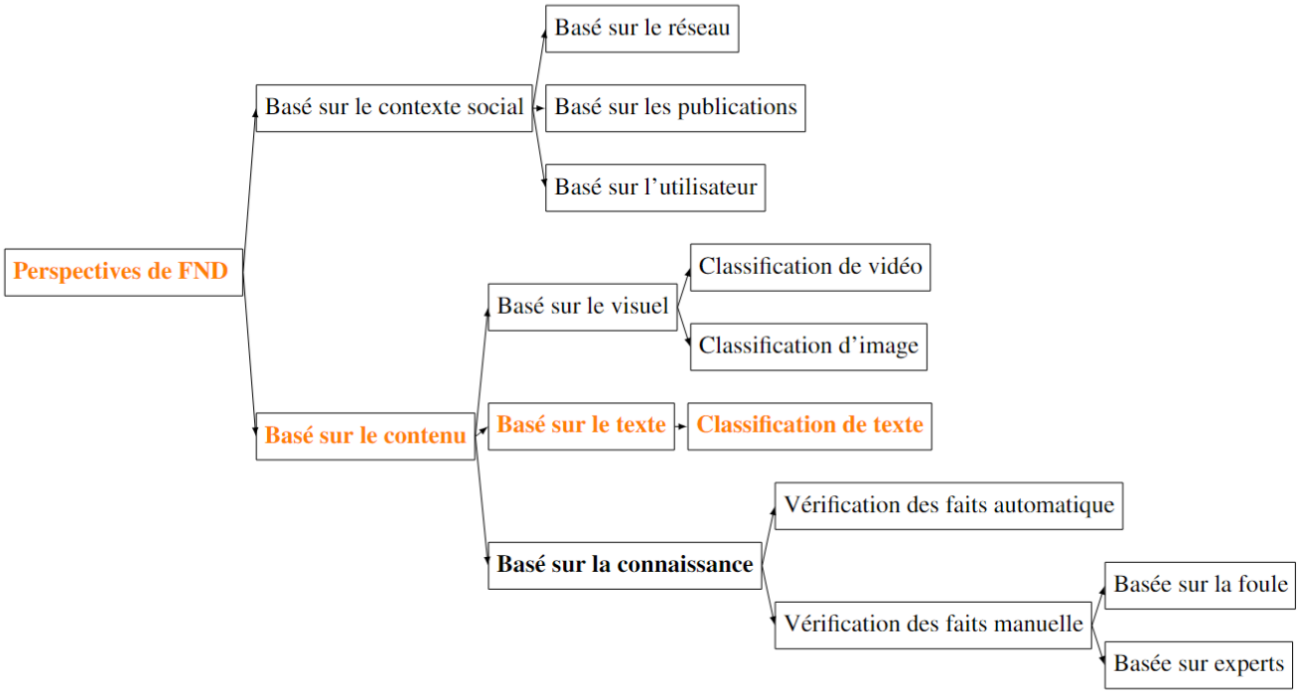


FIGURE 2.1 – Perspectives de détection des fausses nouvelles (FND). Les éléments en orange montrent le focus de cette thése.

2.2 Prétraitement du texte

Le prétraitement du texte est l'étape initiale dans la préparation du texte brut pour le traitement du langage naturel, ce qui aide les machines à comprendre le langage humain. Le but du prétraitement des données est de produire un "texte propre" que les ordinateurs peuvent analyser sans erreurs.Il est essentiel de mettre en place des tâches de prétraitement avant de procéder à l'extraction des caractéristiques et aux étapes d'entraînement, car des données incorrectes peuvent avoir des conséquences néfastes à long terme.

Le prétraitement du texte comprend cinq étapes principales : normalisation, tokenisation, suppression des mots vides, suppression de la ponctuation et racinisation (comme indiqué dans la figure 2.2). Voici des exemples de ces termes :

— Normalisation : Le processus de conversion des termes dans un texte en une forme standard appelée normalisation. Par exemple, "eaaaaaasy" est converti en "easy" ou " :" est converti en "tristesse".

— Tokenisation : Cela signifie le processus de division des données textuelles en unités significatives telles que les mots, les phrases et les documents. Un N-gramme est l'une des méthodes de tokenisation les plus célèbres [14]. Un N-gramme maintient l'ordre des termes dans un document textuel. Il s'agit d'un modèle probabiliste basé sur la théorie de la chaîne de Markov.

— Suppression des mots vides : Processus de suppression des mots ou termes inutiles dans le processus de classification de texte, appelé suppression des mots vides [14]. Dans la langue anglaise, il existe certains mots vides tels que "The", "That", "a", "in", etc.

— Suppression de la ponctuation : Le processus de suppression de certains signes tels que " !", "()", "[]", "’", " ?", "@" etc. des données textuelles est appelé suppression de la ponctuation.

— Stemming : Le processus de conversion d’un mot dans sa forme racine est appelé stemming [14]. Par exemple, les mots "Connected", "Connection", "Connects" et "Connections" sont convertis en "Connect".

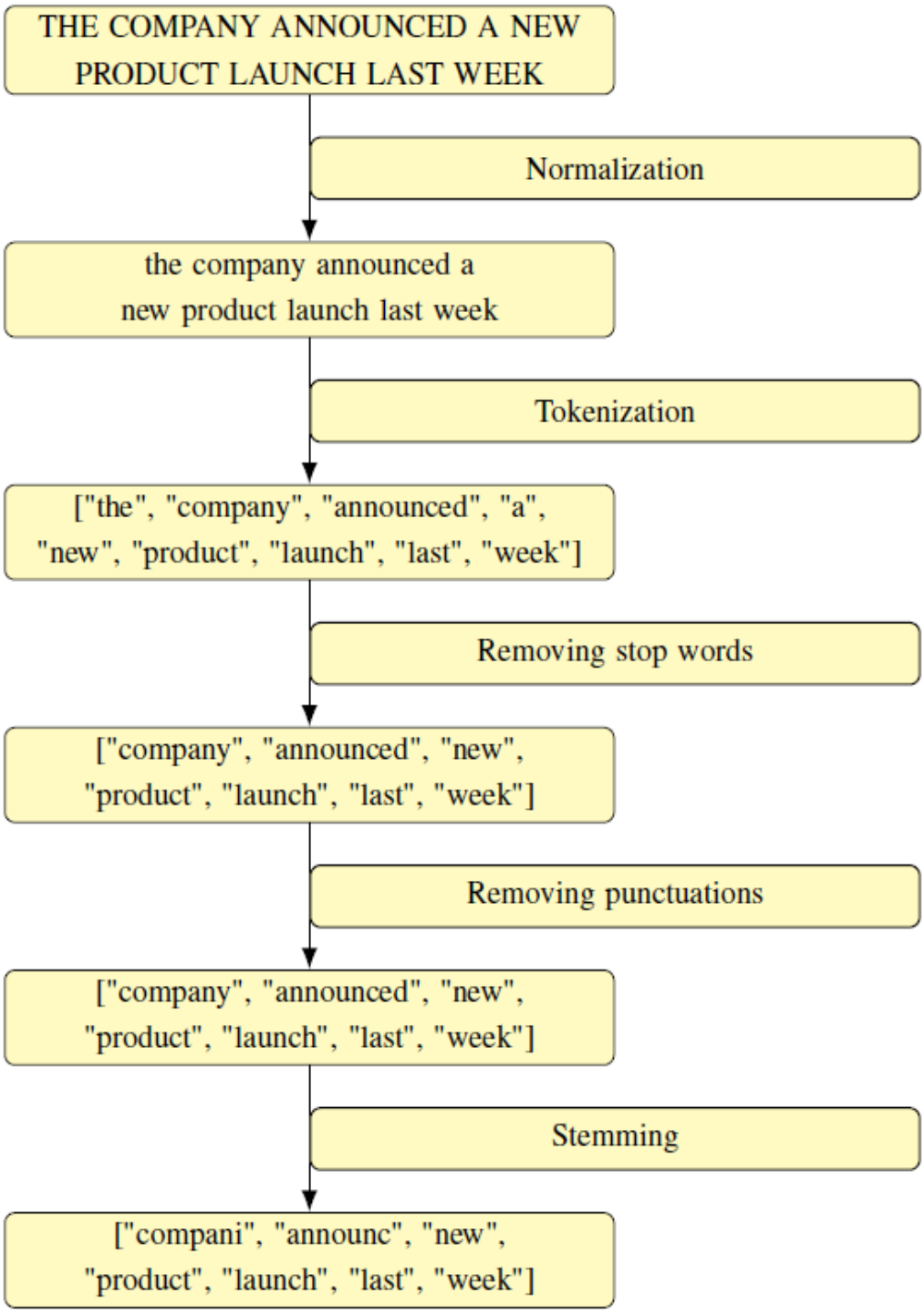


FIGURE 2.2 – Processus de traitement du texte

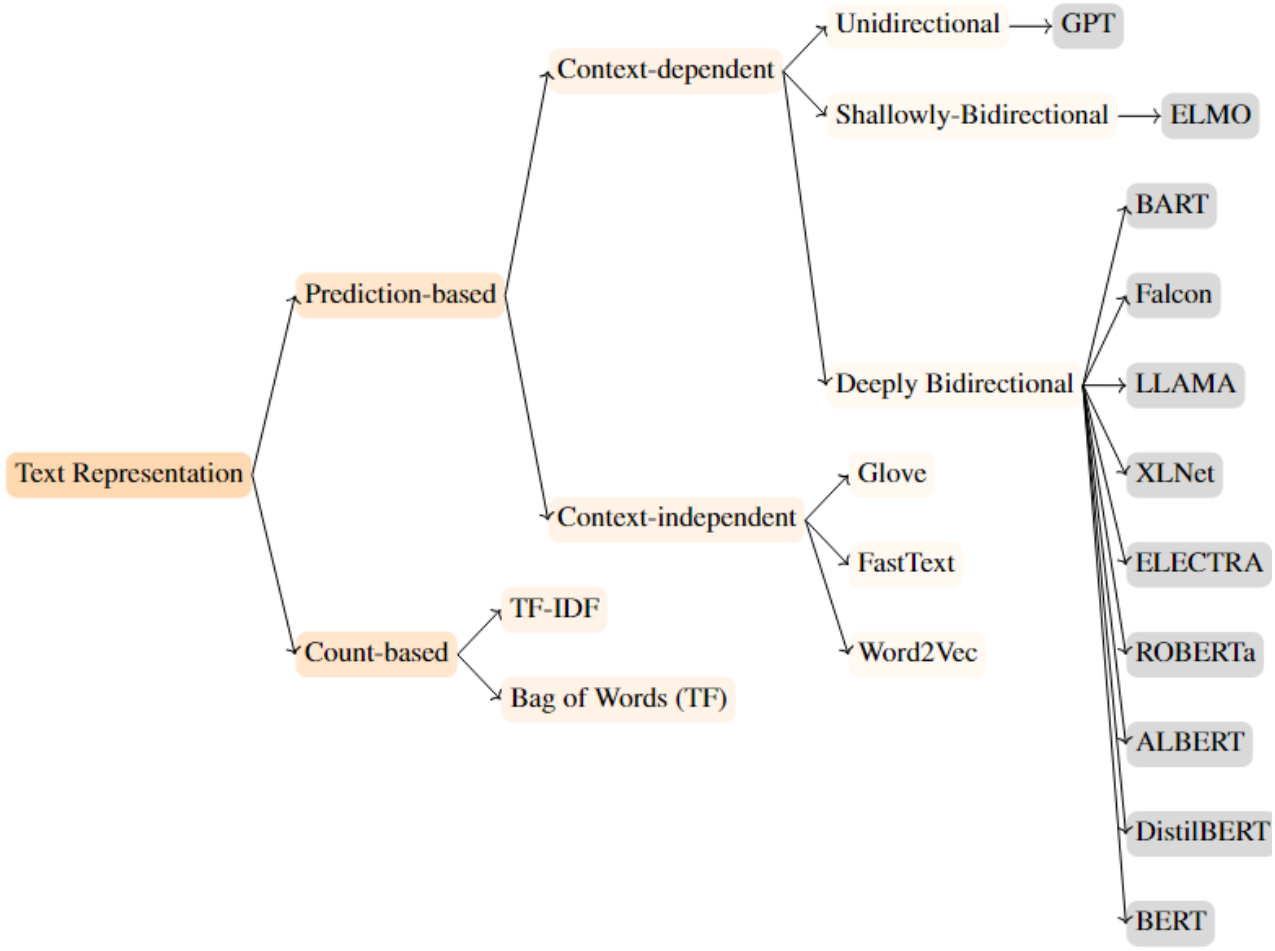


FIGURE 2.3 – Taxonomie des méthodes de représentation de texte.

2.3 Représentation textuelle

La représentation textuelle, également appelée extraction de caractéristiques, est une étape clé dans la classification de texte. Ce processus convertit les données textuelles en vecteurs numériques, qui peuvent être traités par une machine.Il existe plusieurs approches, allant des méthodes simples basées sur le décompte comme le sac de mots (Bag-of-Words) et TF-IDF, aux méthodes plus complexes basées sur la prédiction comme Word2Vec, fastText, ELMO et BERT. Le choix de ces méthodes peut avoir un impact significatif sur les performances des modèles de détection de fausses informations. La figure 2.3 fournit une taxonomie des méthodes de représentation textuelle utilisées dans la détection de fausses informations, tandis que le tableau 2.1 résume les méthodes les plus populaires utilisées dans les travaux antérieurs. Nous passons en revue ces méthodes, leurs avantages et leurs inconvénients, afin de fournir un aperçu des différentes méthodes de représentation textuelle qui peuvent être utilisées dans la détection de fausses informations. Cela aidera les chercheurs et les praticiens à sélectionner la méthode la plus adaptée à leur cas d'utilisation particulier.

TABLE 2.1 – Méthodes d'extraction de caractéristiques utilisées dans les travaux précédents.

Types de caractéristiques	**Ensembles de caractéristiques**	**Techniques**	**Références**
Basé sur le comptage	Basé sur la fréquence	Sac de mots (BOW)	[93] [94]
		TF-IDF	[78, 95] [93, 96] [84, 94]
Basé sur la prédiction	Indépendant du contexte	Word2Vec	[80, 95] [70, 97]
		FastText	[98, 99] [97, 58]
	Dépendant du contexte	GloVe	[100, 99] [97, 58] [85, 101]
		Bidirectional	[102]
		Unidirectional	[103]

2.3.1 Représentation textuelle basée sur le décompte

L'une des méthodes principales de représentation textuelle est la représentation basée sur le décompte, qui se concentre uniquement sur le nombre de fois où les mots apparaissent ou co-apparaissent dans un document. Pour être plus précis, les méthodes basées sur le décompte créent une matrice d'occurrence et de co-occurrence, où chaque élément indique la fréquence d'un mot dans un certain document. Bien que ce type de représentation soit simple et populaire, il présente un inconvénient majeur : Il ne retient aucune donnée importante sur le sens et les liens entre les mots. Les méthodes fréquentielles les plus fréquemment employées sont les suivantes :

Le Sac de Mots

Le Sac de Mots[104] constitue l'une des techniques de représentation textuelle les plus basiques qui ont été employées dans de nombreuses recherches visant à détecter les fausses informations. [105, 106, 96, 107, 70, 84, 98, 94]. Certains documents techniques le désignent comme un vectoriseur de décompte. Cette méthode de représentation décrit simplement l'occurrence et la fréquence des mots (Fréquence des Termes - TF) dans un document. Pour être plus précis, elle ne donne aucune information sur la position et la structure des mots dans un document.Malgré sa simplicité, cette méthode ne donne pas de renseignements sur les relations entre les mots dans les documents. En outre, lorsqu'il s'agit de grandes quantités de données textuelles, la matrice de représentation devient encombrée et demande une grande quantité de puissance de calcul[108]. L'équation (2.2) la formule de manière mathématique. Supposons que W soit une matrice d'occurrence, d désigne un document, t désigne un terme dans un document et V désigne le vocabulaire tel que $t \in V$.

$$W_{dt} = \text{fréquence de } t \text{ dans } d \tag{2.2}$$

TF-IDF

La Fréquence des Termes - Inverse Fréquence des Documents [14] est une méthode de représentation des caractéristiques statistiques en traitement du langage naturel pour scorer et pondérer les mots afin de trouver les mots pertinents et importants dans un document [14]. Cette méthode a été utilisée dans de nombreux problèmes de classification de texte, en particulier pour la détection de fausses informations [79, 93, 105, 106, 94, 36, 109, 82, 110, 111]. Pour le formaliser de manière mathématique, les équations (2.3), (2.4), (2.5), (2.6) le formalisent comme suit :

$$\text{TF}(d,t) = \text{Compt}(terme\ t\ occurrant\ dans\ document\ d) \tag{2.3}$$

$$\text{IDF}(t) = \log\left(\frac{\text{Compt}(documents)}{\text{Compt}(documents\ incluant\ terme\ t)}\right) \tag{2.4}$$

$$\text{TF-IDF}(d,t) = \text{TF}(d,t) \times \text{IDF}(t) \tag{2.5}$$

$$\text{TF-IDF} : \{V \to W | W \in \mathbb{R}^{T \times D}, W_{ij} = \text{TF-IDF}(d_i, t_j)\} \tag{2.6}$$

Contrairement au Sac de Mots (Bag-of-Words), qui fournit simplement une collection de vecteurs basée sur le décompte des occurrences de mots dans le document, le modèle TF-IDF intègre plus que simplement le décompte. Il ajoute des informations sur l'importance de certains termes. Cela confère au TF-IDF un pouvoir discriminatif plus grand, car les mots rares qui sont indicatifs de fausses informations sont pondérés davantage tandis que les mots courants qui ne le sont pas sont pondérés moins. Par conséquent, le TF-IDF peut améliorer la précision de la classification des fausses informations en identifiant des caractéristiques importantes qui pourraient être manquées en utilisant uniquement la fréquence des termes (TF).

2.3.2 Représentation textuelle basée sur la prédiction

Les méthodes d'apprentissage profond ont réalisé des progrès significatifs ces dernières années, notamment dans le domaine de la modélisation du langage. L'une des avancées réside dans l'utilisation de modèles de réseaux neuronaux qui créent des plongements de mots à partir de données textuelles en définissant des tâches telles que la prédiction du mot suivant. Ces méthodes extraient des caractéristiques sémantiques et syntaxiques des mots, répondant aux limitations des méthodes basées sur le décompte. Le premier modèle de langage neuronal utilisant un réseau de neurones à propagation avant a été introduit par [112]. Comme le montre la Fig.2.4, la représentation textuelle par réseau neuronal est divisée

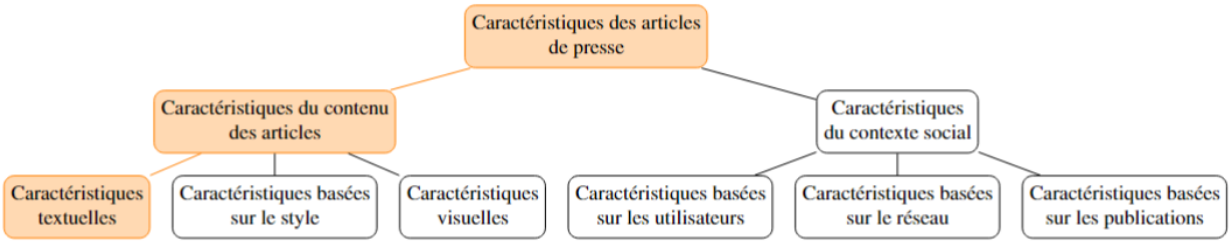


FIGURE 2.4 – Types de caractéristiques dans le problème de détection des fausses nouvelles. Les éléments surlignés en orange montrent le focus de cette recherche.

en deux catégories principales : les méthodes indépendantes du contexte et les méthodes dépendantes du contexte. Dans les méthodes indépendantes du contexte, un mot a toujours la même représentation, quel que soit son contexte. Word2Vec ([113]), Glove ([114]) et fastText [115] sont les méthodes de représentation textuelle indépendantes du contexte les plus populaires. Les méthodes dépendantes du contexte représentent les mots en tenant compte du contexte, divisé en trois classes principales : la représentation unidirectionnelle, la représentation bidirectionnelle peu profonde et la représentation bidirectionnelle profonde du texte. La représentation unidirectionnelle contextualise chaque mot en fonction des mots dans les directions gauche ou droite, tandis que la représentation bidirectionnelle contextualise chaque mot en fonction du contexte à la fois à droite et à gauche ([116]). GPT [134] est la représentation textuelle unidirectionnelle la plus populaire, tandis que BERT ([116]), DistilBERT ([117]), RoBERTa ([118]), BART ([119]), ELECTRA ([120]), XLNet ([121]) et GPT-2 ([122]) sont les méthodes de représentation textuelle bidirectionnelle les plus populaires.

Méthodes indépendantes du contexte

Les modèles indépendants du contexte ou les plongements de mots sont une technique populaire pour représenter les données textuelles sous forme numérique, adaptée aux algorithmes d'apprentissage automatique. Trois méthodes de plongement de mots populaires sont Word2Vec, GloVe et fastText. Cette section propose un aperçu de ces méthodes et met en évidence leurs principales caractéristiques et applications.

Word2Vec a été la première méthode de plongement de mots sans contexte proposée par Mikolov et al. [113] en 2013.Les techniques fréquentielles, comme les techniques de décompte, rencontrent deux problèmes majeurs : elles sont inefficaces lorsque la taille des données augmente et ne tiennent pas compte des similarités entre les mots [123]. Pour surmonter ces limitations, les méthodes sans contexte comme Word2Vec utilisent un modèle de réseau neuronal avec une seule couche cachée pour apprendre la représentation des mots dans les données textuelles. Plus précisément, Word2Vec représente chaque mot sous forme d'un vecteur numérique de longueur fixe de telle sorte que les mots similaires aient des vecteurs de plongement similaires [123]. Deux architectures de modèle de prédiction principales, le sac de mots continu (CBOW) et Skip-gram, sont utilisées dans Word2Vec pour produire des représentations

de mots à partir d'un grand corpus de texte. Les modèles CBOW considèrent à la fois n mots avant et n mots après un mot cible w_t pour prédire le mot cible, tandis que les modèles Skip-gram considèrent un mot cible tel que w_t pour prédire les n mots environnants du mot cible [113]. Word2Vec a été utilisé dans de nombreux problèmes de classification de texte, notamment pour la détection de fausses informations [80, 31, 70, 98, 100]. Cependant, Word2Vec ne peut pas gérer les mots hors vocabulaire en raison du petit contexte local. Néanmoins, la petite zone de contexte rend cette méthode moins coûteuse en termes de mémoire que le modèle GloVe.

GloVe ,qui signifie "global vectors for word embedding" (vecteurs globaux pour l'incrustation de mots), a été publié en 2014 par [114]. Ce modèle extrait directement des informations statistiques globales à partir de mots entiers, comme le terme « Global » le suggère ([114]). GloVe utilise un algorithme non supervisé pour apprendre et représenter la distribution des mots sous forme de vecteurs numériques. Plus précisément, cette méthode utilise la factorisation de matrice pour mapper les mots dans un espace, où la distance entre les mots dans cet espace indique la similarité entre les mots. La factorisation de matrice crée une matrice de cooccurrence de mots pour extraire la relation entre les mots. GloVe a été utilisé dans de nombreux problèmes de classification de texte, notamment pour la détection de fausses nouvelles ([70, 99, 58, 65, 85]). Cependant, GloVe a un coût de mémoire plus élevé que d'autres méthodes comme Word2Vec ou fastText.

FastText est l'une des méthodes d'incrustation de mots les plus puissantes, publiée en 2017 par [115]. Comme Word2Vec, fastText est une représentation textuelle basée sur la prédiction qui utilise le modèle skip-gram avec des modifications mineures dans son architecture pour l'apprentissage de représentations. Plus précisément, le modèle skip-gram de fastText ne prend pas en compte la structure de chaque mot. Au lieu de cela, chaque mot est représenté comme un ensemble de caractères en utilisant la tokenisation par N-grammes [115]. Ainsi, le vecteur d'incrustation de chaque mot est calculé par la somme des vecteurs des caractères N-grammes. Cette méthode a été utilisée dans de nombreux problèmes de classification de texte, notamment pour la détection de fausses nouvelles ([98, 99, 97, 58]). En raison de la représentation au niveau des caractères N-grammes, la représentation fastText surpasse les autres techniques d'incrustation de mots dans les langues morphologiquement riches comme l'arabe ou l'allemand [115]. De plus, comparé à Word2Vec, il peut mieux gérer les mots hors vocabulaire. Cependant, il a un coût de mémoire plus élevé et le nombre optimal de N-grammes doit être ajusté. L'auteur mentionne que le meilleur nombre de N-grammes se situe entre trois et six. Toutefois, cela dépend de la tâche cible et de la langue [115]. Un autre avantage de fastText est qu'il peut également être utilisé pour générer des incrustations de sous-mots, ce qui est utile pour traiter les mots rares ou inconnus.

Méthodes dépendantes du contexte

BERT est une abréviation de Bidirectional Encoder Representations from Transformers, et il s'agit d'une représentation de langage pré-entraînée profondément contextualisée publiée par Devlin et al.

[116], qui a obtenu des résultats significatifs dans de nombreuses tâches de traitement du langage naturel. Le processus de pré-entraînement de BERT se compose de deux principaux modèles : le masquage de langage (Masked Language Modeling, MLM) et la prédiction de la phrase suivante (Next Sentence Prediction, NSP). Le modèle de masquage de langage prédit le pourcentage de jetons d'entrée masqués au hasard [123]. En conséquence, le modèle peut capturer la signification syntaxique et sémantique des mots masqués. Dans la prédiction de la phrase suivante, l'objectif est d'entraîner les relations entre les paires de phrases par une tâche de classification binaire [123]. BERT utilise la tokenisation en morceaux de mots comme entrée des modèles, ce qui aide les modèles à gérer les mots hors vocabulaire ou les mots rares dans un ensemble de données. BERT est pré-entraîné sur un transformateur de 12 à 24 couches sur de grandes quantités de données non structurées provenant de Wikipedia et de livres. Un manque de jeux de données annotés, en particulier ceux spécifiques à une tâche, est le plus grand problème dans les tâches de traitement du langage naturel. Le principal avantage des modèles pré-entraînés profondément contextualisés comme BERT est qu'ils sont des systèmes bidirectionnels non supervisés. Cela signifie qu'ils sont pré-entraînés sur un grand nombre de données non annotées et qu'ils peuvent être ajustés pour toute tâche en NLP. Malgré le fait que les transformateurs aient résolu bon nombre des insuffisances des méthodes de représentation de texte précédentes et aient obtenu des résultats significatifs sur une variété de tâches en NLP, ils ont un processus d'apprentissage lent et ne sont pas rentables en raison du grand nombre de paramètres entraînables. Plus précisément, le modèle BERT-base comporte plus de 110 millions de paramètres entraînables.

ALBERT est une abréviation de A Lite BERT, publié en 2019 par Lan et al. [124]. ALBERT est une version légère de l'apprentissage auto-supervisé de la représentation textuelle de BERT. L'augmentation de la taille des modèles de transformateurs améliore leurs performances dans toutes les tâches NLP en aval. Cependant, cette augmentation de la taille des modèles ralentit le processus d'entraînement et entraîne des coûts de mémoire [124]. Pour résoudre ce problème, ils ont utilisé l'architecture BERT avec quelques différences clés. Premièrement, les paramètres de plongement de mots sont factorisés en les divisant en deux matrices plus petites [124]. Deuxièmement, grâce au partage de paramètres entre les couches, les paramètres de couche sont partagés pour chaque sous-segment comparable, ce qui entraîne une diminution considérable du nombre de paramètres. Ainsi, l'échange de paramètres permet non seulement de réduire le coût computationnel de l'entraînement, mais rend également l'entraînement plus efficace [124]. Troisièmement, la prédiction de l'ordre des phrases, souvent appelée SOP, est utilisée à la place de la perte de prédiction de la phrase suivante (NSP) pour mesurer la cohérence inter-phrases [124]. Même si ALBERT a 70% de paramètres en moins que le modèle BERT, il obtient souvent des performances supérieures. De plus, les modèles ALBERT surpassent les modèles BERT en termes de capacité de données et peuvent être entraînés 1,7 fois plus rapidement [124].

BART ([119]) est une abréviation de Bidirectional Auto-Regressive Transformers, publié en 2019 par Lewis et al. BART est une combinaison de BERT et GPT en raison du fait qu'il est bidirectionnel comme

BERT et auto-régressif comme le modèle pré-entraîné GPT. BART est entraîné sur un modèle séquence-à-séquence avec un encodeur bidirectionnel et un décodeur auto-régressif ([119]). Pendant le processus d'entraînement, le texte est d'abord corrompu en utilisant un générateur de bruit aléatoire, puis un modèle pour réassembler le texte original est appris. Il utilise une architecture typique de traduction automatique neuronale basée sur Transformer ([119]).

Le tableau 2.2 présente une comparaison des avantages et des inconvénients de divers modèles de détection des fausses nouvelles basés sur l'apprentissage automatique. La méthode TF-IDF, bien que capable de distinguer les mots significatifs des moins importants, se révèle lente pour de grands vocabulaires et ne tient pas compte des positions, sémantiques et co-occurrences des mots ([125], [126]). Le modèle Sac de mots est simple à implémenter mais ignore l'ordre et les relations sémantiques des mots ([127], [128]). Word2Vec préserve la signification sémantique et contextuelle des mots, mais ne gère pas les mots inconnus et manque de représentations sous-mot ([85], [129]). Doc2Vec offre une représentation numérique rapide des documents, mais est moins efficace pour les textes courts ([130], [131]). GloVe, contrairement à Word2Vec, utilise des statistiques globales pour les vecteurs de mots, se basant sur la co-occurrence des mots ([85], [84]). Enfin, BERT excelle dans la capture de la signification contextuelle à partir de grandes quantités de données, mais est computationnellement intensif et sensible au bruit en l'absence de prétraitement ([132], [133], [134]).

2.4 Algorithmes de classification

Dans cette section, nous allons décrire l'aspect théorique des modèles d'apprentissage automatique et d'apprentissage profond que nous avons utilisés pour l'analyse afin de fournir un aperçu de ces technologies et méthodes algorithmiques. La section contiendra également des éléments mathématiques et/ou du pseudocode qui expliqueront la complexité des modèles d'apprentissage profond et d'apprentissage automatique. Étant donné que le problème porte sur la classification de texte, nous n'expliquerons que les modèles qui s'appliquent à ce type de problème.

Nous présentons ici les algorithmes de classification les plus populaires utilisés dans la détection de fausses informations. Ces algorithmes ont été choisis en fonction de leur importance dans les travaux précédents, présentés dans le Tableau 2. En tenant compte de cela, nous avons catégorisé les algorithmes de classification en trois grandes catégories : les modèles d'apprentissage automatique classiques, les modèles d'apprentissage ensembliste, et les modèles d'apprentissage profond (Fig.2.5).

2.4.1 Machine Learning : Apprentissage Automatique

Chaque fois que nous consultons un dossier médical, que nous analysons des résultats de tests, que nous suivons l'évolution des patients, ou que nous collectons des données cliniques, nous générons des

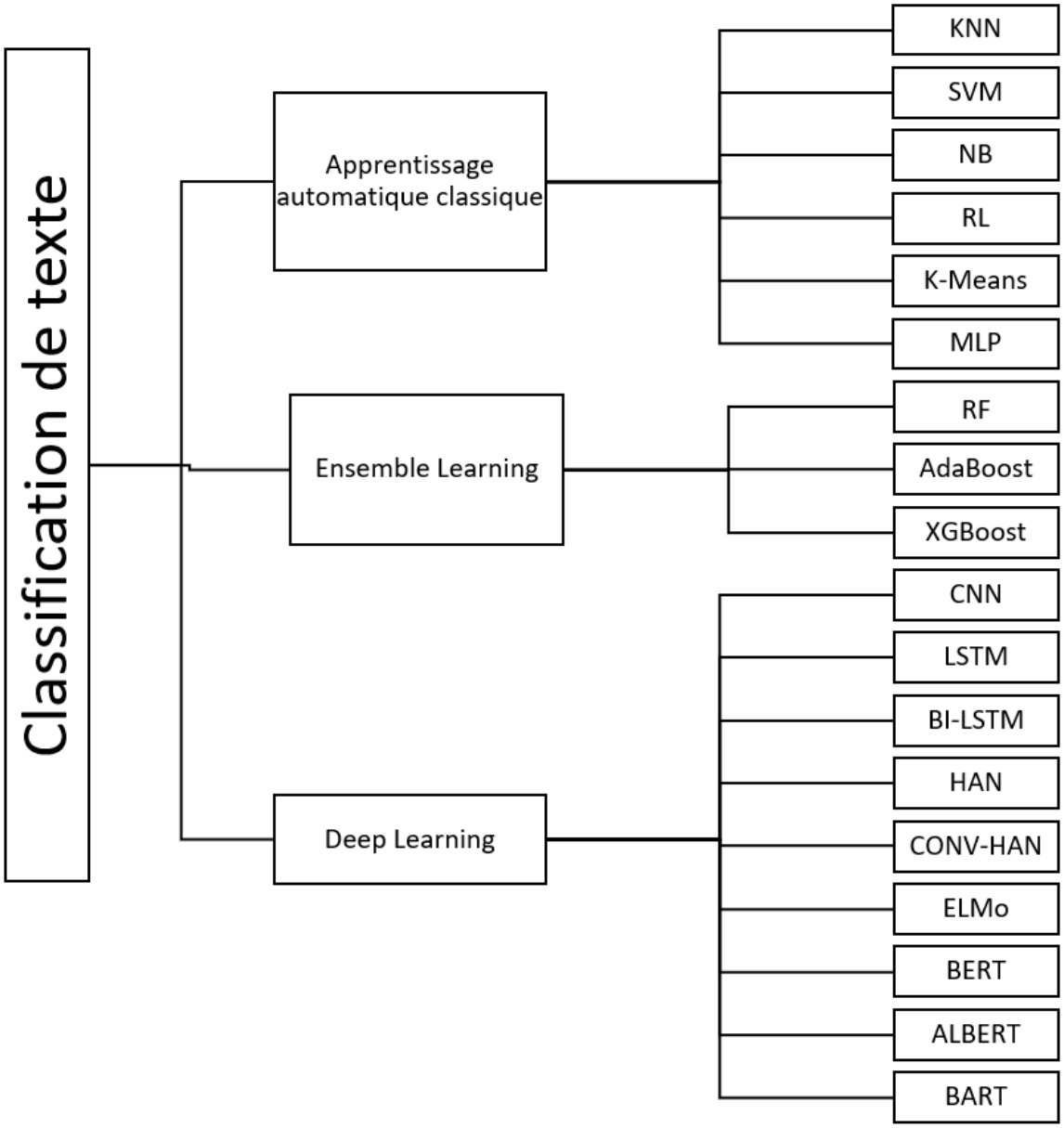


FIGURE 2.5 – Classification des approches de classification textuelle.

TABLE 2.2 – Avantages et inconvénients de certains modèles de détection des fausses nouvelles basés sur l'apprentissage automatique

Méthode	**Avantages**	**Inconvénients**	**Référence**
TF-IDF	Le modèle TF-IDF inclut des informations sur les mots les plus significatifs et les moins importants	Lent pour les grands vocabulaires. Ne capture pas la position dans le texte, les sémantiques, et les co-occurrences dans différents documents	[125],[126]
Sac de mots	Facile à implémenter	Ignore l'ordre des mots dans un document et ignore également les relations sémantiques entre les mots	[127], [128]
Word2Vec	Maintient la signification sémantique de divers mots dans le texte et l'information contextuelle est également préservée.	Incapable de gérer les mots inconnus. Il n'y a pas de représentations communes au niveau des sous-mots	[85], [129]
Doc2Vec	Plus rapide que Word2Vec et une représentation numérique d'un document indépendamment de sa longueur	L'avantage de Doc2Vec est diminué pour les documents plus courts	[130], [131]
GloVe	Contrairement à Word2Vec, il ne se base pas uniquement sur des statistiques locales (informations contextuelles locales des mots)	Pour obtenir des vecteurs de mots, des statistiques globales (co-occurrence des mots) sont utilisées	[85], [84]
BERT	Identifie et capture la signification contextuelle dans une phrase ou un texte, notamment à partir de grandes quantités de données.	Intensif en calcul lors de l'inférence. De plus, il n'intègre pas de prétraitement des données, ce qui peut impacter la performance en présence de bruit.	[132], [133], [134]

données. Les professionnels de la santé sont non seulement des générateurs mais aussi des consommateurs de données. Nous souhaitons avoir des diagnostics et des traitements spécialisés pour chaque patient. Nous voulons que les besoins des patients soient compris et que leurs traitements soient prédits avec précision. Prenons, par exemple, un hôpital qui gère des milliers de dossiers de patients à travers plusieurs services spécialisés et cliniques. Les détails de chaque consultation sont enregistrés : date, identifiant du patient, symptômes observés, traitements administrés, résultats des tests, etc. Cela représente généralement une grande quantité de données chaque jour. Ce que l'hôpital veut, c'est pouvoir prédire quel traitement est le plus susceptible d'être efficace pour quel patient, afin de maximiser les chances de guérison et d'améliorer la qualité des soins. De même, chaque patient souhaite recevoir le traitement qui

correspond le mieux à ses besoins spécifiques.

Le comportement des patients change avec le temps et selon des facteurs par exemple, l'âge, le style de vie et l'historique médical. Mais nous savons qu'il n'est pas complètement aléatoire. Il existe certains schémas dans les données médicales. Pour résoudre un problème de diagnostic sur un ordinateur, nous avons besoin d'un algorithme. Un algorithme est une séquence d'instructions qui doivent être exécutées pour transformer l'entrée en sortie. Cependant, certaines tâches ne peuvent pas être résolues simplement en appliquant le bon algorithme car, parfois, nous ne savons pas comment transformer les données d'entrée pour produire une sortie logique. Pour surmonter ce problème, nous essayons de trouver une approximation appropriée et utile qui transforme (ou explique) nos données de la meilleure manière possible. Cette méthode de recherche de la meilleure approximation/estimation appropriée est appelée Apprentissage Automatique.

Définition 2 : L'apprentissage automatique est le processus de recherche des schémas les plus utiles qui décrivent le mieux nos données, toujours en termes d'approximation.

Divers algorithmes ont été créés, cependant, tout algorithme d'apprentissage automatique qui existe et n'a pas été utilisé ne sera pas couvert dans ce travail. Notre approche, puisqu'elle concerne la classification de texte, se compose d'algorithmes appartenant au groupe des algorithmes d'apprentissage supervisé.

Modèles d'apprentissage automatique classiques

Les algorithmes d'apprentissage automatique classiques couvrent un large éventail d'algorithmes comprenant des modèles probabilistes (par exemple, Naive Bayes, Régression Logistique), des modèles basés sur la mémoire (par exemple, K-plus proches voisins, Machine à vecteurs de support), et des modèles basés sur des règles (par exemple, Arbre de Décision).

SVM est un algorithme d'apprentissage supervisé particulièrement efficace pour la classification binaire 1 et est largement utilisé dans le domaine de la détection de fausses informations. En utilisant l'algorithme SVM, il est possible de déterminer l'hyperplan optimal qui divise les données en deux catégories différentes, telles que les informations vérifiées et les informations fausses . Cela est réalisé à l'aide des vecteurs de support, qui sont des instances proches de l'hyperplan et influencent sa position et son orientation. La classification est effectuée en fonction de la position d'une instance par rapport à l'hyperplan. L'efficacité de SVM réside dans sa capacité à maximiser la marge entre les deux classes, En d'autres termes, il s'agit de la distance entre l'hyperplan et les points de données les plus proches de chaque côté, ce qui réduit le risque de mauvaise classification concernant les nouveaux échantillons. Pour améliorer sa performance dans des espaces de grande dimension, SVM utilise des fonctions noyaux (kernels) telles que le noyau linéaire, polynomial, ou radial (RBF, Radial Basis Function), qui permettent de transformer les données d'entrée non linéaires en un espace de plus haute dimension où un hyperplan linéaire peut être utilisé pour la séparation([96, 93, 36, 109, 82, 100, 135, 136]). Le modèle SVM est généralement une alternative robuste pour la classification de textes car il peut facilement gérer une

quantité limitée de données et des espaces de haute dimension. De plus, il peut obtenir de bonnes performances de généralisation grâce au concept de marge maximale. Ainsi, de nombreuses études ont utilisé SVM pour la détection des fausses nouvelles ([79, 80, 31, 93, 106]). Par exemple, une recherche menée par [10] a démontré que SVM, combiné à des techniques avancées de représentation de texte comme les word embeddings, peut atteindre des niveaux de précision élevés dans la détection de fausses informations sur des plateformes telles que Twitter et Facebook. Dans le cadre de la détection de fausses informations, les SVM sont souvent utilisés en conjonction avec des techniques de traitement du langage naturel (NLP) pour analyser le texte des articles ou des publications sur les réseaux sociaux. Par exemple, les caractéristiques textuelles telles que la fréquence des mots (TF-IDF), la présence de certaines phrases ou termes caractéristiques de fausses informations, et les traits syntaxiques peuvent être extraites et utilisées comme vecteurs d'entrée pour l'algorithme SVM ([137, 138]. En somme, le modèle SVM, avec ses capacités de classification puissante et son aptitude à gérer des données textuelles complexes, constitue un outil précieux pour la détection de fausses informations, permettant ainsi de filtrer efficacement les contenus véridiques des contenus fallacieux dans le flot incessant d'informations sur internet.

Algorithm 1 Algorithme des Machines à Vecteurs de Support

function TRAINSVM(training_data, labels, C, kernel)
    Initialize $\alpha_i = 0$ for all i ▷ Lagrange multipliers
    Initialize $b = 0$ ▷ Threshold parameter
    while not converged **do**
        for each i in training_data **do**
            Calculate $E_i = f(x_i) - y_i$ ▷ Error for current example
            if $(y_i \cdot E_i < -\varepsilon$ and $\alpha_i < C)$ or $(y_i \cdot E_i > \varepsilon$ and $\alpha_i > 0)$ **then**
                Select $j \neq i$ randomly
                Calculate $E_j = f(x_j) - y_j$
                Save old α values : $\alpha_i^{old} = \alpha_i$, $\alpha_j^{old} = \alpha_j$
                Compute bounds L and H
                if $L == H$ **then**
                    continue to next i
                Compute $\eta = 2 \cdot K(x_i, x_j) - K(x_i, x_i) - K(x_j, x_j)$
                if $\eta \geq 0$ **then**
                    continue to next i
                Update $\alpha_j : \alpha_j = \alpha_j - \frac{y_j(E_i - E_j)}{\eta}$
                Clip α_j to be within bounds L and H
                if $|\alpha_j - \alpha_j^{old}| < \varepsilon$ **then**
                    continue to next i
                Update $\alpha_i : \alpha_i = \alpha_i + y_i \cdot y_j \cdot (\alpha_j^{old} - \alpha_j)$
                Compute b_1 and b_2 using updated α values
                Update b to be the average of b_1 and b_2
    return α, b
function PREDICT(test_data, $\alpha, b, kernel$) **for** *eachxintest_data* **do**
    function $(f)(x) = \sum \alpha_i y_i K(x_i, x) + b$
        IF $f(x) \geq 0$ **THEN RETURN** 1 **ELSE RETURN** -1

Naives Bayes Le modèle Naïve Bayes est un algorithme de classification utilisé pour des problèmes de classification binaire (deux classes) et multi-classes 2. La technique est plus facile à comprendre lorsqu'elle est décrite en utilisant des valeurs d'entrée binaires ou catégorielles. Naïve Bayes est basé sur le théorème de Bayes qui stipule que :

$$P(h|d) = \frac{P(d|h) \cdot P(h)}{P(d)} \tag{2.7}$$

Cela signifie que :

— $P(h|d)$ est la probabilité de l'hypothèse h étant donnée les données d. C'est ce qu'on appelle la probabilité *a posteriori*.

— $P(d|h)$ est la probabilité des données d étant donné que l'hypothèse h est vraie.

— $P(h)$ est la probabilité que l'hypothèse h soit vraie (indépendamment des données). C'est ce qu'on appelle la probabilité *a priori* de h.

— $P(d)$ est la probabilité des données (indépendamment de l'hypothèse).

Le modèle Naïve Bayes est une méthode d'apprentissage supervisé largement utilisée pour les tâches de classification, y compris la détection de fausses informations. Ce modèle repose sur le théorème de Bayes et l'hypothèse "naïve" selon laquelle les caractéristiques (ou attributs) des données sont conditionnellement indépendantes les unes des autres, compte tenu de la classe. Cette simplification permet de construire des modèles probabilistes robustes et efficaces même avec des ensembles de données de grande dimension.

Dans le contexte de la détection de fausses informations, le modèle Naïve Bayes peut être utilisé pour classifier les journaux, les publications sur les plateformes sociales, ou d'autres formes de contenu textuel en tant que vraies ou fausses. Le processus commence par l'extraction des caractéristiques textuelles, telles que les mots ou les phrases, qui sont ensuite utilisées pour estimer les probabilités conditionnelles nécessaires au modèle. Par exemple, les fréquences de mots, les n-grammes, et les scores TF-IDF (Term Frequency-Inverse Document Frequency) sont couramment utilisés comme caractéristiques d'entrée ([105, 106]).

Plusieurs études ont démontré l'efficacité du modèle Naïve Bayes dans la détection de fausses informations. Par exemple, une étude réalisée par [62] a montré que les modèles Naïve Bayes, lorsqu'ils sont combinés avec des techniques de traitement du langage naturel avancées, peuvent atteindre des niveaux de précision compétitifs dans la détection de fake news. En outre, le modèle est apprécié pour sa simplicité, sa rapidité d'entraînement et sa capacité à gérer des ensembles de données volumineux avec une complexité computationnelle relativement faible.

Cependant, cette méthode est limitée par l'hypothèse naïve et ne peut donc pas modéliser les interactions entre les mots dans les données textuelles. Malgré cette limitation, le modèle Naïve Bayes calcule la probabilité qu'un document appartienne à une classe donnée (vrai ou faux) en multipliant les probabilités conditionnelles des caractéristiques observées par la probabilité a priori de la classe. L'algorithme sélectionne ensuite la classe ayant la plus haute probabilité a posteriori comme la prédiction finale. Cette approche probabiliste est particulièrement efficace pour les tâches de classification textuelle, car elle peut gérer de manière efficace les caractéristiques discrètes et continues ([98, 109, 139, 103]).

En somme, le modèle Naïve Bayes est un outil précieux pour la détection de fausses informations, offrant une approche probabiliste simple mais efficace pour classifier les contenus textuels. Grâce à sa capacité à exploiter les caractéristiques textuelles de manière indépendante et à générer des prédictions rapides et fiables, il constitue un composant essentiel dans l'arsenal des techniques de détection de fake news.

Algorithm 2 Algorithme de Naïve Bayes pour la Détection des Fausses Nouvelles

for each class c in classes **do** ▷ loop for each class
    $P(c) = \frac{\text{count}(c)}{N}$ ▷ calculate prior probability for class c
for each feature f in features **do** ▷ loop for each feature
    for each value v in $f.values$ **do** ▷ loop for each value of feature f
        for each class c in classes **do** ▷ loop for each class
            $P(f = v|c) = \frac{\text{count}(f=v \text{ and } c)}{\text{count}(c)}$ ▷ calculate likelihood of feature value given class
function PREDICT(instance) ▷ function to predict class of a new instance
    for each class c in classes **do** ▷ loop for each class
        $posterior[c] = P(c)$ ▷ initialize posterior probability with prior
        for each feature f in instance.features **do** ▷ loop for each feature in the instance
            $value = instance[f]$ ▷ get the feature value
            $posterior[c] * = P(f = value|c)$ ▷ multiply with the likelihood of the feature value given class
    $predicted\_class = \text{argmax}(posterior)$ ▷ select the class with the highest posterior probability
    return $predicted\_class$
=0

KNN Le modèle des K-plus proches voisins est un modèle basé sur la mémoire qui détermine la classe d'une nouvelle instance en fonction de la classe des instances les plus proches dans l'espace des caractéristiques. Il existe quelques études qui ont utilisé KNN pour la détection des fausses nouvelles car il s'agit d'un apprenant paresseux . KNN est très sensible à la fois aux caractéristiques non pertinentes et à la taille des ensembles de données. De plus, il peut nécessiter beaucoup de ressources informatiques et une grande mémoire car il stocke toutes les données d'entraînement.

Algorithm 3 Algorithme des K Plus Proches Voisins pour la Détection des Fausses Nouvelles

```
function TRAINKNN(training_data, labels)
    Store training_data and labels
function PREDICT(test_data, training_data, labels, k)
    predictions = []
    for each x_i in test_data do
        distances = []
        for each (x_j, y_j) in training_data do
            distance = EuclideanDistance(x_i, x_j)
            distances.append((distance, y_j))
        Sort distances by the first element (distance)
        neighbors = distances[ :k]                    ▷ select the k-nearest neighbors
        votes = { }
        for each (distance, label) in neighbors do
            if label not in votes then
                votes[label] = 0
            votes[label] += 1
        predicted_label = argmax(votes)               ▷ select the label with the most votes
        predictions.append(predicted_label)
    return predictions
function EUCLIDEANDISTANCE(point1, point2)
    sum = 0
    for each feature f in point1 do
        sum += (point1[f] - point2[f])^2 −
        return sqrt(sum)
```

La régression logistique est un modèle de classification extrêmement simple à mettre en œuvre, mais qui possède une excellente performance sur des classes linéairement séparables. Il s'agit d'un des algorithmes les plus couramment employés dans le domaine de la classification. La régression logistique est nommée en référence à la fonction logistique utilisée au cœur de la méthode. Les statisticiens ont élaboré la fonction logistique, aussi connue sous le nom de fonction sigmoïde, afin de décrire les caractéristiques de la croissance des populations en écologie, qui augmentent rapidement et atteignent un niveau maximal de charge pour l'environnement. Il s'agit d'une courbe en forme de S qui peut prendre un nombre réel et le convertir en une valeur comprise entre 0 et 1, mais jamais précisément à ces limites. (algorithm. 4).

La régression logistique est l'un des classificateurs probabilistes. Elle est similaire au modèle de régression, sauf que la variable de résultat doit être catégorielle. Nous pouvons utiliser ce modèle sous trois

formats : régression logistique binomiale, régression logistique multinomiale et régression logistique ordinale. Certaines études ont utilisé la régression logistique pour la détection de fausses informations ([31, 93, 106, 96, 98, 94, 36, 111, 135]).

Algorithm 4 Algorithme de Régression Logistique pour la Détection des Fausses Nouvelles

function TRAINLOGISTICREGRESSION(training_data, labels, learning_rate, epochs)
    Initialize weights $w = [w_0, w_1, ..., w_n]$ to small random values
    Initialize bias b to a small random value
    for epoch = 1 to epochs **do** ▷ loop for the number of training iterations
        for each (x_i, y_i) in training_data **do**
            $z = w \cdot x_i + b$ ▷ linear combination of inputs and weights
            $\hat{y} = \frac{1}{1+e^{-z}}$ ▷ apply the sigmoid function
            $error = \hat{y} - y_i$
            $w = w - learning\_rate \cdot error \cdot x_i$ ▷ update weights
            $b = b - learning\_rate \cdot error$ ▷ update bias
    return w, b
function PREDICT(test_data, weights, bias)
    for each x_i in test_data **do**
        $z = weights \cdot x_i + bias$
        $\hat{y} = \frac{1}{1+e^{-z}}$
        if $\hat{y} \geq 0.5$ **then**
            return "Fake"
        else
            return "Real"

Modèles ensemblistes

Un modèle ensembliste est un système d'apprentissage qui utilise plusieurs classificateurs de base pour obtenir de meilleures performances prédictives [140]. L'objectif d'un ensemble est d'obtenir une meilleure précision que n'importe quel classificateur individuel, car les points forts d'un modèle peuvent compenser les faiblesses des autres. En d'autres termes, ils se complètent et conduisent à des prédictions plus robustes. Plusieurs travaux ont utilisé des modèles ensemblistes pour la détection de fausses informations ([79, 80, 107, 84, 98, 94, 83, 141, 100]). Les méthodes ensemblistes les plus couramment utilisées dans la détection de fausses informations sont basées sur des variantes de la méthode Bagging (par exemple, les forêts aléatoires [142]) et le Boosting [143].

Modèle de Forêt Aléatoire est un algorithme d'apprentissage supervisé développé par [142], basé sur l'algorithme de bagging. Cet algorithme sélectionne un bootstrap des données d'entraînement pour former l'ensemble d'entraînement de chaque modèle de base individuel. Comme chaque modèle de

base est entraîné avec un bootstrap différent (c'est-à-dire un échantillonnage avec remplacement), ils aboutissent à un ensemble de modèles diversifiés à la fin, ce qui aide à réduire la variance du modèle ([142]).

Le modèle de forêt aléatoire est une implémentation particulière de cet algorithme où les caractéristiques et les échantillons sont échantillonnés aléatoirement avec remplacement pour générer un ensemble d'arbres diversifiés. Ensuite, il fait une prédiction basée sur le vote majoritaire des résultats de prédiction de tous les arbres de décision. La forêt aléatoire améliore les performances en réduisant la variance. Plusieurs études ont utilisé la forêt aléatoire pour la détection de fausses informations ([79, 107, 83]) En construisant une "forêt" d'arbres de décision,lors de la croissance des arbres, la forêt aléatoire apporte une randomisation supplémentaire au modèle. Plutôt que de chercher la caractéristique la plus cruciale lors de la division d'un nœud, elle cherche la meilleure caractéristique parmi un ensemble aléatoire de caractéristiques. Cela conduit à une grande variété qui conduit généralement à un modèle amélioré. Ainsi, dans une forêt aléatoire, l'algorithme ne prend en considération qu'un sous-ensemble aléatoire des caractéristiques pour diviser un nœud, ce qui améliore la précision et la stabilité des prédictions.

Algorithm 5 Algorithme de Forêt Aléatoire pour la Détection des Fausses Nouvelles

```
function TRAINRANDOMFOREST(training_data, labels, num_trees, max_features)
    Initialize forest to an empty list
    for each tree t from 1 to num_trees do
        bootstrap_sample ← BootstrapSample(training_data, labels)
        tree ← TrainDecisionTree(bootstrap_sample, max_features)
        Add tree to forest
    return forest
function BOOTSTRAPSAMPLE(data, labels)
    Initialize sample_data and sample_labels to empty lists
    for each i from 1 to len(data) do
        index ← random integer between 1 and len(data)
        Add data[index] to sample_data
        Add labels[index] to sample_labels
    return (sample_data, sample_labels)
function TRAINDECISIONTREE(data, labels, max_features)
    Create a decision tree with the specified max_features
    Train the decision tree on the data and labels
    return trained decision tree
function PREDICT(forest, test_data)
    Initialize predictions to an empty list
    for each x in test_data do
        Initialize votes to an empty list
        for each tree in forest do
            prediction ← PredictWithTree(tree, x)
            Add prediction to votes
        final_prediction ← MajorityVote(votes)
        Add final_prediction to predictions
    return predictions
function PREDICTWITHTREE(tree, x)
    return prediction of x using tree
function MAJORITYVOTE(votes)
    return the most common element in votes
```

2.5 Modèles d'apprentissage profond

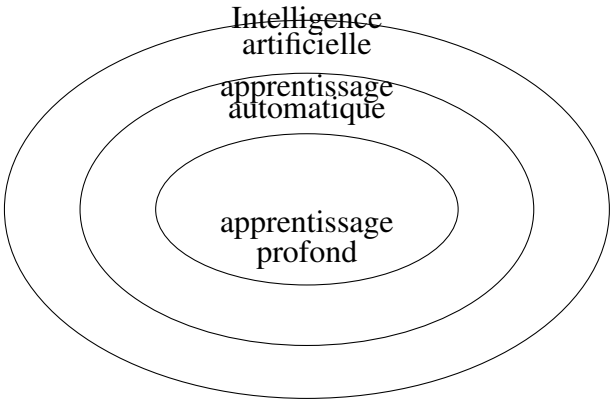


FIGURE 2.6 – Intelligence artificielle, apprentissage automatique, apprentissage profond([2])

Les modèles d'apprentissage profond sont un type de machine learning et d'intelligence artificielle qui ne nécessite pas d'extraction manuelle des caractéristiques. Au lieu de cela, ils apprennent automatiquement les représentations des caractéristiques à partir des données brutes. Les modèles d'apprentissage profond ont montré des résultats prometteurs dans de nombreuses tâches et sont de plus en plus utilisés dans la classification de texte. Cependant, ils nécessitent également plus de ressources informatiques que les modèles de machine learning traditionnels. Plusieurs études ont examiné l'utilisation des modèles d'apprentissage profond pour la détection de fausses informations. Dans cette section, nous décrivons certains des modèles d'apprentissage profond les plus populaires qui ont été utilisés pour la détection de fausses informations.

2.6 Principe de Base des Réseaux de Neurones Artificiels

Un réseau neuronal biologique normal est constitué d'un ensemble et d'un groupe de neurones fonctionnellement ou chimiquement similaires. Chaque neurone individuel est généralement connecté à un grand nombre d'autres neurones, ce qui rend le nombre total de connexions et de neurones dans le réseau relativement élevé. Les synapses se forment lorsque les axones se lient aux dendrites, bien que des synapses dendrodendritiques et d'autres types de connexions soient également envisageables. D'autres formes de signalisation émergent de la diffusion des neurotransmetteurs en plus de la transmission électrique [144].

D'autre part, le réseau neuronal (NN) est connu pour être un processeur parallèle distribué massif constitué d'unités de base ayant une capacité et une tendance naturelle à stocker et à rendre accessibles des informations d'expérience. Il s'agit généralement d'une technique de calcul conçue pour imiter le cerveau dans la résolution de problèmes. Le cerveau humain et les réseaux neuronaux sont similaires en ce sens qu'ils apprennent à analyser les entrées et à résoudre des problèmes par l'apprentissage.

Les réseaux neuronaux peuvent également être appelés réseaux neuronaux stimulés (SNN) et réseaux neuronaux artificiels (ANN) [144].

Aujourd'hui, les réseaux neuronaux sont utilisés dans plusieurs applications telles que la robotique, la reconnaissance faciale humaine, les applications médicales, la reconnaissance vocale, l'économie et la fabrication.

2.6.1 Architecture des Systèmes de Réseaux Neuronnaux

Le cerveau humain est principalement constitué de 10^{11} unités de traitement appelées neurones, qui travaillent ensemble en parallèle. En plus d'échanger des informations par l'intermédiaire de leurs synapses, qui sont des unités de connexion, ces neurones additionnent principalement toutes les informations qu'ils reçoivent. Si le résultat dépasse le potentiel d'action donné, un signal est envoyé directement via l'axone vers la phase suivante [144].

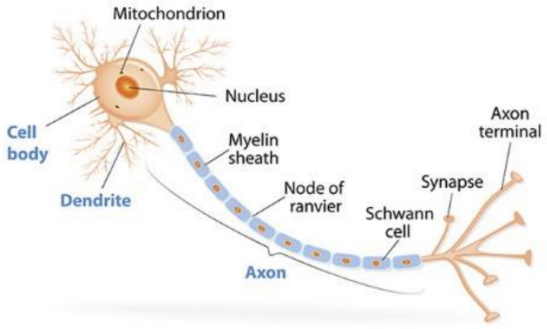


FIGURE 2.7 – L'anatomie des neurones humains

Comme dans le cerveau humain, les réseaux neuronaux artificiels sont également composés de petites unités simples, appelées neurones artificiels. Chaque unité est liée aux autres unités par des connecteurs de poids. Ensuite, ces unités utilisent une fonction d'activation pour calculer la somme pondérée des entrées reçues et déterminer la sortie.

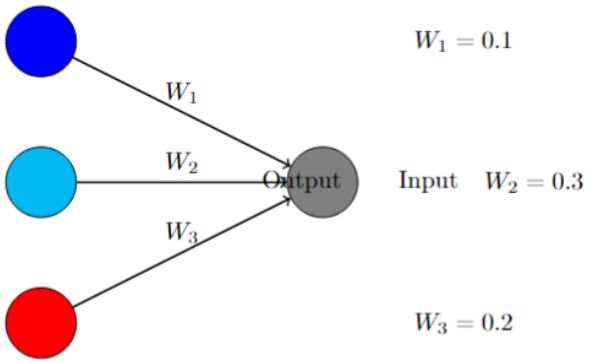


FIGURE 2.8 – Un neurone artificiel de base

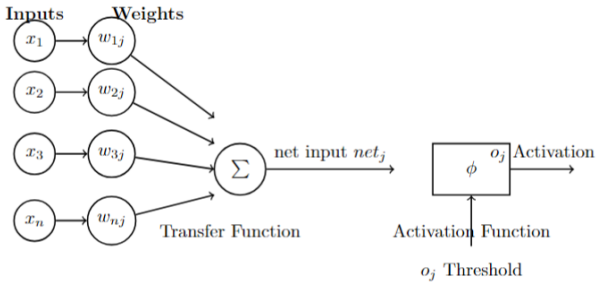


FIGURE 2.9 – Un diagramme en blocs d'un neurone artificiel

Plus le poids d'un neurone artificiel est élevé, plus l'entrée amplifiée par le neurone artificiel est grande. Le signal est atténué par le poids négatif, qui peut également être négatif. Le calcul du neurone variera en fonction des poids [144]). Quatre aspects essentiels d'un modèle neuronal ont été identifiés sur la base du diagramme en blocs (2.9) et de la fonction du réseau neuronal :

— Les liens de connexion ou synapses ont une certaine force ou poids, où le signal de l'entrée X_i connecté et lié au neurone k est multiplié par le poids synaptique w_{ki}.

— L'addition des entrées pondérées.

— La sortie d'un neurone est produite par une fonction d'activation. Elle est également connue sous le nom de fonction d'écrasement car elle réduit (limite) la plage d'amplitude du signal de sortie à une valeur limitée.

— Selon le biais b_k, qui est un nombre positif ou négatif, cela influence l'effet d'augmentation ou de réduction de la somme nette des entrées de la fonction d'activation [144].

CNN

La détection des fausses nouvelles est une tâche cruciale à l'ère des médias sociaux où les fausses nouvelles peuvent se propager rapidement et avoir des conséquences graves dans le monde réel. Les CNN se sont révélés efficaces pour détecter les fausses nouvelles en raison de leur capacité à apprendre et à extraire automatiquement les caractéristiques pertinentes des données textuelles. Les modèles basés sur les CNN ont été appliqués avec succès à diverses tâches de détection de fausses nouvelles, telles que l'identification d'articles de fausses nouvelles, la détection de titres de fausses nouvelles et la détection de faux avis. Plusieurs études ont rapporté des résultats prometteurs en utilisant des modèles basés sur les CNN dans la détection des fausses nouvelles.

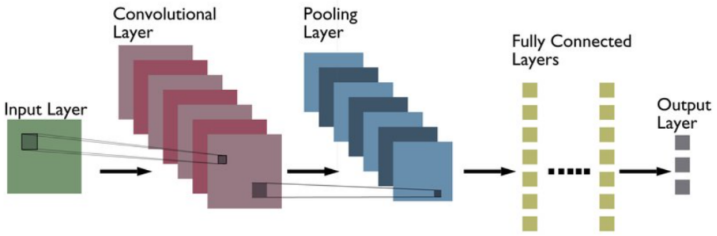


FIGURE 2.10 – Modèle CNN [3]

L'un des principaux défis dans la détection des fausses nouvelles est l'absence de grands ensembles de données annotées. Pour relever ce défi, certains chercheurs ont proposé de tirer parti des techniques d'apprentissage par transfert pour utiliser des modèles de langage pré-entraînés, tels que BERT ou GPT, afin d'améliorer les performances des modèles basés sur les CNN [145],[146].

D'autres chercheurs se sont concentrés sur la conception de nouvelles architectures CNN, comme le montre la Figure 2.10, pour la détection des fausses nouvelles. Par exemple, Huang et al. ont proposé un modèle CNN à double canal qui incorpore des embeddings au niveau des caractères et des mots pour capturer différents niveaux d'information dans le texte [147]. De même, Qiu et al. ont introduit un modèle hybride CNN-LSTM qui combine les forces des deux architectures [148]. Le modèle hybride utilise une CNN pour extraire les caractéristiques locales du texte et une LSTM pour capturer les dépendances séquentielles entre les mots.

Un autre aspect important de la détection des fausses nouvelles est l'identification des caractéristiques pertinentes. Les chercheurs ont exploré diverses approches pour la sélection des caractéristiques, y compris l'utilisation des tags de partie du discours (POS) [149], l'analyse des sentiments [150], et la reconnaissance des entités nommées (NER). Ces caractéristiques peuvent être intégrées dans les modèles basés sur les CNN pour améliorer leurs performances.

En fait, certains chercheurs ont dirigé leur attention vers l'atténuation des attaques adversariales sur les modèles de détection de fausses nouvelles basés sur les CNN. Les attaques adversariales impliquent l'ajout de perturbations imperceptibles aux données d'entrée pour tromper le modèle. Pour contrer ce défi, les chercheurs ont suggéré d'utiliser la formation adversariale [151] et le réseau d'attention hiérarchique dans les modèles basés sur les CNN.

Dans l'ensemble, les modèles basés sur les CNN ont montré des résultats prometteurs dans la détection des fausses nouvelles, et les recherches futures se concentreront probablement sur le développement de modèles plus robustes capables de gérer les attaques adversariales et de mieux capturer les nuances du langage humain. Par exemple, Zhang et al. [152] ont introduit un modèle basé sur les CNN utilisant des embeddings de mots et des couches de convolution pour classifier les articles de presse comme étant soit faux, soit réels. Le modèle a démontré une précision de 92,8% sur un ensemble de données d'articles de presse.

Dans une autre étude, Yang et al. [153] ont présenté un modèle basé sur les CNN qui intègre à la fois des informations textuelles et visuelles pour identifier les titres de fausses nouvelles. Ce modèle a

obtenu un score F1 de 0,794 sur un ensemble de données de titres de fausses et vraies nouvelles. De plus, Wang et al. [154] ont introduit un modèle basé sur les CNN qui intègre des caractéristiques linguistiques et le réseau d'attention hiérarchique pour détecter les faux avis. Le modèle a démontré une précision de 92,8% sur un ensemble de données comprenant à la fois des avis faux et réels.

Une des approches populaires pour la détection des fausses nouvelles basée sur les CNN consiste à utiliser une combinaison de couches de convolution et de max pooling pour capturer les caractéristiques locales et leurs interactions à travers différentes régions du texte d'entrée. Par exemple, dans le travail de Kim [155], un modèle CNN a été entraîné sur un grand ensemble de données d'articles de presse et de messages sur les réseaux sociaux pour identifier les fausses nouvelles. Le modèle était composé de plusieurs couches de convolution et de pooling, suivies d'une couche entièrement connectée pour la classification. Les résultats ont montré que le modèle CNN proposé surpassait d'autres méthodes dans la détection des fausses nouvelles.

Une autre approche pour la détection des fausses nouvelles basée sur les CNN consiste à utiliser des embeddings de mots pré-entraînés pour représenter le texte d'entrée et à utiliser plusieurs filtres de différentes tailles pour capturer différents niveaux de granularité dans le texte d'entrée. Par exemple, de même que Ma et al. [156], un modèle CNN a été entraîné sur un ensemble de données d'articles de presse et de tweets pour détecter les fausses nouvelles. Le modèle utilisait des embeddings de mots pré-entraînés et plusieurs filtres de différentes tailles pour capturer les caractéristiques locales et leurs interactions. Les résultats ont montré que le modèle CNN proposé atteignait une haute précision dans la détection des fausses nouvelles.

À tel point que certains chercheurs ont également exploré l'utilisation du réseau d'attention hiérarchique dans les CNN pour la détection des fausses nouvelles. Par exemple, dans le travail de Zhang et al. [152], un modèle CNN avec attention a été proposé pour détecter les fausses nouvelles à partir des plateformes de médias sociaux. Le mécanisme d'attention attribuait des poids à différents mots dans le texte d'entrée en fonction de leur pertinence pour la tâche de détection des fausses nouvelles. Les résultats ont montré que le modèle CNN proposé surpassait d'autres méthodes pour détecter les fausses nouvelles.

Yang et al. [157] proposent un CNN multicanal pour la détection des fausses nouvelles qui combine des embeddings de caractères, de mots et de documents. Les auteurs mènent des expériences sur un ensemble de données à grande échelle et montrent que leur modèle surpasse plusieurs bases de référence, y compris les modèles d'apprentissage automatique traditionnels et d'autres architectures de réseaux neuronaux. Wang et al. [158] présentent un modèle hybride qui combine un CNN et un LSTM bidirectionnel pour la détection des fausses nouvelles. Le CNN est utilisé pour extraire des caractéristiques des embeddings de mots, et le LSTM bidirectionnel est utilisé pour capturer des informations contextuelles. Les auteurs évaluent leur modèle sur un ensemble de données de référence et montrent qu'il surpasse plusieurs bases de référence.

Zhang et al. [152] proposent une approche basée sur les CNN pour la détection des fausses nouvelles en utilisant des données de microblogs. Les auteurs utilisent des embeddings de mots pré-entraînés et un CNN pour apprendre les caractéristiques locales et globales du texte. Ils évaluent leur modèle sur un

ensemble de données de microblogs chinois et montrent qu'il surpasse plusieurs bases de référence.

Ghosh et al. [159] explorent l'utilisation de caractéristiques multimodales et de CNN pour la détection des fausses nouvelles. Les auteurs utilisent à la fois des caractéristiques textuelles et visuelles pour entraîner un modèle basé sur les CNN. Ils évaluent leur modèle sur un ensemble de données de référence et montrent que l'incorporation de caractéristiques visuelles peut améliorer les performances.

Ainsi, Wang et al. [160] proposent un CNN hiérarchique avec attention pour la détection des fausses nouvelles. Les auteurs utilisent une architecture CNN à deux niveaux pour capturer à la fois les caractéristiques locales et globales du texte, puis appliquent l'attention pour mettre en évidence les caractéristiques importantes. Ils évaluent leur modèle sur un ensemble de données de référence et montrent qu'il surpasse plusieurs bases de référence.

Algorithm 6 Algorithme de Réseau de Neurones Convolutif pour la Détection des Fausses Nouvelles

function TRAINCNNMODEL(training_data, labels, num_epochs, batch_size, learning_rate, num_filters, filter_sizes, dropout_rate, embedding_dim, vocab_size, max_seq_length)
    Initialize CNN model with specified parameters
    for each epoch from 1 to num_epochs **do**
        for each batch in training_data divided by batch_size **do**
            $X\_batch, y\_batch \leftarrow$ GetNextBatch(training_data, labels, batch_size)
            predictions $\leftarrow$ ForwardPass(CNN, X_batch)
            loss $\leftarrow$ ComputeLoss(predictions, y_batch)
            gradients $\leftarrow$ BackwardPass(loss)
            UpdateWeights(CNN, gradients, learning_rate)
    return Trained CNN model
function PREDICT(CNN_model, test_data)
    Initialize predictions to an empty list
    for each x in test_data **do**
        prediction $\leftarrow$ ForwardPass(CNN_model, x)
        Add prediction to predictions
    return predictions
function FORWARDPASS(CNN_model, X)
    embeddings $\leftarrow$ Embed(X, vocab_size, embedding_dim)
    conv_outputs $\leftarrow$ []
    for each filter_size in filter_sizes **do**
        conv_output $\leftarrow$ Conv1D(embeddings, filter_size, num_filters)
        relu_output $\leftarrow$ ReLU(conv_output)
        pooled_output $\leftarrow$ MaxPooling1D(relu_output)
        Add pooled_output to conv_outputs
    concatenated_output $\leftarrow$ Concatenate(conv_outputs)
    flattened_output $\leftarrow$ Flatten(concatenated_output)
    dense_output $\leftarrow$ Dense(flattened_output, units, activation='relu')
    dropout_output $\leftarrow$ Dropout(dense_output, dropout_rate)
    final_output $\leftarrow$ Dense(dropout_output, num_classes, activation='softmax')
    return final_output
function COMPUTELOSS(predictions, labels)
    loss $\leftarrow$ CategoricalCrossentropy(predictions, labels)
    return loss
function BACKWARDPASS(loss)
    gradients $\leftarrow$ ComputeGradients(loss)
    return gradients
function UPDATEWEIGHTS(CNN_model, gradients, learning_rate)
    ApplyGradients(CNN_model, gradients, learning_rate)
    return
function EMBED(X, vocab_size, embedding_dim)
    embeddings $\leftarrow$ EmbeddingLayer(X, vocab_size, embedding_dim)
    return embeddings

RNN

Une forme de Réseau de Neurones Récurrents appelée Long Short-Term Memory est créée pour résoudre le problème du gradient en disparition qui affecte les RNN conventionnels. Le RNN peut avoir du mal à apprendre des dépendances à long terme en raison du problème du gradient en disparition, qui survient lorsque le gradient utilisé pour mettre à jour les poids lors de la rétropropagation diminue trop pour effectuer des ajustements significatifs sur les poids.[161]

Pour résoudre ce problème, les LSTM incluent une cellule de mémoire et trois mécanismes de porte responsables de la gestion du flux d'informations entrant et sortant de la cellule. Les trois portes sont la porte d'entrée, la porte d'oubli et la porte de sortie. La porte d'entrée régule la quantité de nouvelles informations pouvant entrer dans la cellule, la porte d'oubli régule la quantité d'informations pouvant quitter la cellule, et la porte de sortie régule la quantité d'informations pouvant passer de la cellule à la sortie. Un composant crucial des LSTM est la cellule de mémoire, qui permet au réseau de stocker et de récupérer sélectivement des données au fil du temps. Chaque fois que les mécanismes de porte sont activés, une combinaison linéaire de l'état précédent de la cellule et de l'entrée actuelle est utilisée pour mettre à jour la cellule. La porte d'entrée choisit quelles nouvelles informations sont ajoutées à l'état de la cellule, la porte d'oubli choisit quelles informations sont conservées de l'état précédent de la cellule, et la porte de sortie choisit quelles informations sont transférées à la couche suivante du réseau.

Les LSTM ont été largement utilisés pour des activités de traitement du langage naturel (NLP) comme la traduction automatique, l'analyse des sentiments et la classification de texte, ainsi que pour des applications supplémentaires comme la reconnaissance vocale, la description d'images et l'analyse vidéo. Le LSTM est une sorte de réseau neuronal récurrent qui s'adapte bien aux entrées temporelles ou séquentielles telles que les textes. Un RNN est un type de réseau neuronal où l'état caché est alimenté dans une boucle avec les entrées séquentielles. Il y en a généralement une version déroulée (Figure 2.11). Chacun des X_i étant une valeur dans la séquence.

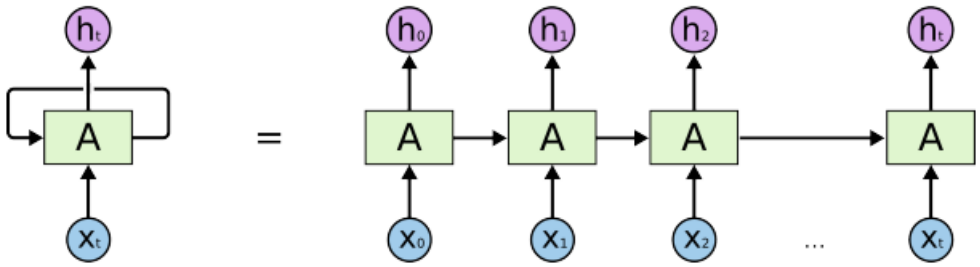


FIGURE 2.11 – Un réseau de neurones récurrent déroulé ([4]).

Dans ce cas, les valeurs X_i sont des vecteurs de mots. Il y a deux possibilités, soit utiliser un vecteur pré-entraîné avec word2vec, soit faire de X_i un paramètre à apprendre de la même manière que cela fonctionne pour l'algorithme Word2Vec, en ayant un encodage one-hot du mot et une matrice de poids à régler. Chaque méthode sera utilisée. Les réseaux de neurones récurrents ne fonctionnent pas très bien

avec les dépendances à long terme, c'est pourquoi les LSTM ont été introduits. Il est composé d'une porte d'entrée, d'une porte de sortie et d'une porte d'oubli qui sont combinées dans l'équation suivante 2.8.

$$f_t = \sigma_g(\mathbf{W}_f \mathbf{x}_t + \mathbf{U}_f \mathbf{h}_{t-1} + \mathbf{b}_f) \tag{2.8}$$

La figure 2.12 montre comment cela fonctionne. Un LSTM bidirectionnel fonctionne de la même manière, mais l'entrée est alimentée dans les deux sens, du début à la fin et de la fin au début.

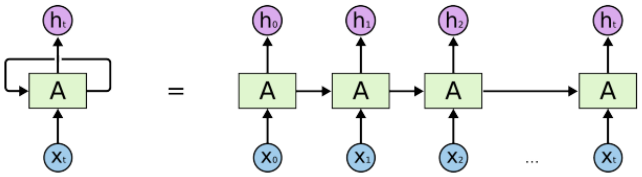


FIGURE 2.12 – LSTM gates([4])

2.6.2 BI-LSTM

Les modèles de mémoire à long terme bidirectionnels (Bi-LSTM) sont un type d'architecture de réseau neuronal récurrent (RNN) couramment utilisés pour le traitement des données séquentielles, notamment les tâches de traitement du langage naturel (NLP) [162]. Le modèle Bi-LSTM étend le modèle LSTM traditionnel en prenant en compte les informations des étapes passées et futures de la séquence d'entrée. Dans le Bi-LSTM, la séquence d'entrée est traitée dans deux directions : avant et arrière. Cela signifie que la séquence d'entrée est alimentée dans deux réseaux LSTM distincts. L'un traite la séquence dans son ordre original (LSTM avant) et l'autre traite la séquence dans l'ordre inverse (LSTM arrière). Les sorties LSTM avant et arrière sont ensuite concaténées à chaque étape temporelle.

— État de la cellule :

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W_c x_t + W_c h_{t-1} + b_c) \tag{2.9}$$

— Porte d'entrée :

$$i_t = \sigma(W_{ix} x_t + W_{ih} h_{t-1} + b_i) \tag{2.10}$$

— Porte de sortie :

$$o_t = \sigma(W_{ox} x_t + W_{oh} h_{t-1} + b_o) \tag{2.11}$$

— État caché :

$$h_t = o_t \cdot \tanh(c_t) \tag{2.12}$$

— Porte d'oubli :

$$f_t = \sigma(W_{fx} x_t + W_{fh} h_{t-1} + b_f) \tag{2.13}$$

Les cellules Bi-LSTM comportent de nombreuses couches pour chaque itération T, y compris une couche d'entrée X_t, une couche de sortie h_t et une couche cachée h_{t-1}. Chaque cellule partage certains états avec d'autres cellules pendant l'entraînement ou les mises à jour des paramètres, comme illustré à la Figure 2.13.

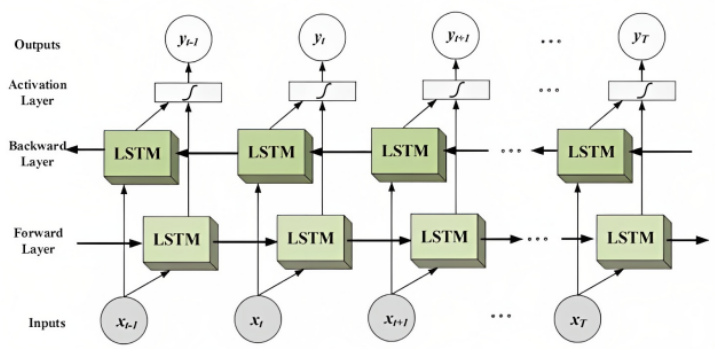


FIGURE 2.13 – Modèle Bi-LSTM [5]

William Yang Wang et al.[163] propose une méthode pour détecter les fausses informations sur les réseaux sociaux en utilisant des réseaux neuronaux LSTM bidirectionnels. Les auteurs ont construit un ensemble de données de 16 000 tweets en anglais contenant des informations vraies et fausses et ont atteint une précision de 85% avec leur modèle BiLSTM.

Monti et al.[164] ont proposé une méthode pour repérer les fausses nouvelles sur les réseaux sociaux en utilisant des réseaux neuronaux convolutifs et des réseaux BiLSTM. Ils ont comparé leur approche à d'autres méthodes existantes de détection de désinformation et ont démontré que leur modèle était plus efficace. Ils ont également utilisé une méthode de géométrie d'apprentissage profond pour améliorer la précision.

Jia et al.[165] ont proposé une méthode pour détecter les fausses nouvelles en utilisant des réseaux neuronaux BiLSTM, en tenant compte de la position de l'auteur par rapport aux faits présentés dans le texte. Les auteurs ont utilisé un ensemble de données contenant des articles de presse en anglais et ont démontré que leur modèle était plus efficace que d'autres approches de détection de désinformation.

Tan et al.[166] ont proposé un BiLSTM avec un mécanisme d'attention pour détecter les fausses nouvelles. Le modèle a été entraîné sur un ensemble de données d'articles de presse étiquetés comme réels ou faux et a atteint des performances de pointe par rapport à d'autres modèles sur le même ensemble de données.

L'étude utilisée dans [167] utilise un BiLSTM avec un mécanisme d'auto-attention pour détecter les fausses nouvelles sur les réseaux sociaux en intégrant les informations de contexte social. Le modèle a atteint des performances de pointe sur un ensemble de données de tweets étiquetés comme réels ou faux.

Zhang et al.[152] ont proposé un BiLSTM basé sur l'attention pour détecter les fausses nouvelles qui peut capturer à la fois les dépendances locales et globales dans le texte. Le modèle a été évalué sur deux ensembles de données d'articles de presse étiquetés comme réels ou faux et a atteint des performances

de pointe par rapport à d'autres modèles sur les mêmes ensembles de données. Cette étude a proposé un BiLSTM basé sur l'attention pour détecter les rumeurs sur les réseaux sociaux en utilisant l'apprentissage multitâche, qui consiste à apprendre conjointement à classifier les rumeurs et à identifier la source de la rumeur. Le modèle a atteint des performances de pointe sur un ensemble de données de tweets étiquetés comme rumeurs ou non-rumeurs.

Feng et al.[168] ont proposé un BiLSTM avec attention graphique pour détecter les fausses nouvelles dans les articles de presse. Le modèle incorpore les relations entre les entités dans le texte en utilisant une structure de graphe et a atteint des performances de pointe sur un ensemble de données d'articles de presse étiquetés comme réels ou faux.

Goyal et al.[169] ont proposé un modèle de détection des fausses nouvelles qui combine BiLSTM avec un mécanisme d'attention et CNN. Le modèle proposé a atteint un score F1 de 0,94 sur l'ensemble de données LIAR, surpassant plusieurs modèles de pointe.

Wu et al.[170] ont proposé un modèle de détection des fausses nouvelles qui combine BiLSTM avec CNN et un mécanisme d'attention. Le modèle proposé a atteint un score F1 de 0,91 sur l'ensemble de données LIAR, surpassant plusieurs modèles de pointe.

Panigrahi et al.[171] ont proposé un modèle de détection des fausses nouvelles qui intègre BiLSTM avec CNN et un mécanisme d'attention, atteignant un score F1 remarquable de 0,89 sur l'ensemble de données LIAR. De même, Qian et al.[172] ont proposé un modèle de détection des fausses nouvelles qui combine BiLSTM avec CNN et un mécanisme d'attention. Le modèle proposé a atteint un score F1 de 0,83 sur l'ensemble de données LIAR, surpassant plusieurs modèles de pointe.

Dans une autre étude de Zhang et al.[152], les auteurs ont utilisé un modèle BiLSTM-CNN pour la détection des fausses nouvelles. Le modèle tire parti à la fois de la nature séquentielle et convolutive des données d'entrée pour extraire des caractéristiques pertinentes. Les auteurs ont évalué leur modèle sur deux ensembles de données de référence et ont obtenu des résultats de pointe. De même, une étude d'Arora et al.[149] a proposé un modèle basé sur BiLSTM qui utilise des embeddings contextuels et un mécanisme d'attention pour la détection des fausses nouvelles. Les auteurs ont évalué leur modèle sur un ensemble de données d'articles de presse et ont atteint une précision de 92,4%.

Dans une étude récente de Chen et al.[173], les auteurs ont proposé un modèle multitâche basé sur BiLSTM pour la détection des fausses nouvelles. Le modèle effectue deux tâches simultanément : identifier la véracité des articles de presse et détecter leur position. Les auteurs ont évalué leur modèle sur un ensemble de données d'articles de presse politiques et ont obtenu des résultats de pointe sur les deux tâches.

Qiu et al.[148] décrit un système basé sur Bi-LSTM pour reconnaître les caractères manuscrits en utilisant l'ensemble de données NIST. Les auteurs prétraitent les données en utilisant des techniques de normalisation d'image et d'augmentation des données, et utilisent un réseau Bi-LSTM pour classifier les caractères. Ils atteignent une précision de 97,3% sur l'ensemble de données NIST.

Sun et al.[174] décrit un système d'apprentissage profond pour la reconnaissance de mots anglais manuscrits en utilisant le dataset NIST. Les auteurs utilisent un réseau Bi-LSTM pour traiter les images

de mots et atteignent une précision de 93,6%. Ils comparent également les performances de leur système à d'autres approches, y compris les méthodes basées sur des caractéristiques traditionnelles et d'autres modèles d'apprentissage profond.

Zhao et al.[161] décrit un système basé sur Bi-LSTM pour la reconnaissance de mots anglais manuscrits en utilisant le dataset NIST. Les auteurs utilisent une combinaison de techniques d'augmentation de données, y compris la rotation et la translation, pour augmenter la taille de l'ensemble d'entraînement. Ils atteignent une précision de 95,5% sur le dataset NIST et comparent leur approche à d'autres modèles d'apprentissage profond.

Conneau dans son article[175] compare les performances des réseaux de neurones convolutionnels (CNN) et des réseaux Bi-LSTM sur une gamme de tâches de classification de texte. Les auteurs constatent que les modèles CNN surpassent les modèles Bi-LSTM sur la plupart des tâches, mais que les modèles Bi-LSTM performent mieux sur certaines tâches où il est important de capturer les dépendances à long terme dans le texte.

Kim et al.[176] propose un modèle CNN pour la classification de phrases et compare ses performances à des modèles traditionnels tels que les modèles bag-of-words et n-gram. L'auteur constate que le modèle CNN surpasse les autres modèles sur plusieurs datasets de référence.

Zhang et al.[152] analysent les performances des CNN, des réseaux Bi-LSTM et d'autres modèles pour la classification de phrases à travers une gamme d'hyperparamètres et de paramètres d'entrée. Les auteurs constatent que les réseaux Bi-LSTM performent bien sur les petits datasets, tandis que les CNN performent mieux sur les grands datasets. Ils fournissent également des recommandations pour optimiser les performances des deux modèles.

Tang et al.[177] proposent un modèle de réseau de neurones récurrents avec portes (GRU) pour la classification des sentiments et comparent ses performances à des modèles traditionnels tels que les machines à vecteurs de support et les arbres de décision. Les auteurs constatent que le modèle GRU surpasse les autres modèles sur plusieurs datasets de référence. Ces articles montrent que les réseaux de neurones BiLSTM sont une approche prometteuse pour la détection des fausses nouvelles et la vérification des faits. Il est possible d'utiliser ces outils seuls ou en collaboration avec d'autres méthodes de traitement du langage naturel afin d'améliorer l'efficacité de la détection des fausses informations.

Algorithm 7 Algorithme Bi-LSTM pour la Détection des Fausses Nouvelles

```
function TRAINBILSTMMODEL(training_data, labels, num_epochs, batch_size, learning_rate, embedding_dim, vocab_size, max_seq_length, lstm_units, dropout_rate)
    Initialize BI-LSTM model with specified parameters
    for each epoch from 1 to num_epochs do
        for each batch in training_data divided by batch_size do
            X_batch, y_batch ← GetNextBatch(training_data, labels, batch_size)
            predictions ← ForwardPass(BI-LSTM, X_batch)
            loss ← ComputeLoss(predictions, y_batch)
            gradients ← BackwardPass(loss)
            UpdateWeights(BI-LSTM, gradients, learning_rate)
    return Trained BI-LSTM model
function PREDICT(BILSTM_model, test_data)
    Initialize predictions to an empty list
    for each x in test_data do
        prediction ← ForwardPass(BILSTM_model, x)
        Add prediction to predictions
    return predictions
function FORWARDPASS(BILSTM_model, X)
    embeddings ← Embed(X, vocab_size, embedding_dim)
    bi_lstm_output ← BI-LSTM(embeddings, lstm_units)
    dropout_output ← Dropout(bi_lstm_output, dropout_rate)
    final_output ← Dense(dropout_output, num_classes, activation='softmax')
    return final_output
function COMPUTELOSS(predictions, labels)
    loss ← CategoricalCrossentropy(predictions, labels)
    return loss
function BACKWARDPASS(loss)
    gradients ← ComputeGradients(loss)
    return gradients
function UPDATEWEIGHTS(BILSTM_model, gradients, learning_rate)
    ApplyGradients(BILSTM_model, gradients, learning_rate)
    return
function EMBED(X, vocab_size, embedding_dim)
    embeddings ← EmbeddingLayer(X, vocab_size, embedding_dim)
    return embeddings
```

2.6.3 HAN

HAN est un type de modèle d'apprentissage profond conçu pour gérer les structures hiérarchiques dans les données textuelles. Le modèle HAN comprend deux niveaux d'attention : l'attention au niveau des mots et l'attention au niveau des phrases. La couche d'attention au niveau des mots attribue des poids différents à chaque mot de la séquence, en fonction de son importance dans le contexte de l'article de presse. La couche d'attention au niveau des phrases attribue des poids différents à chaque phrase de l'article, en fonction de son importance pour déterminer si l'article est réel ou faux. Plusieurs études récentes ont exploré l'utilisation de HAN pour la détection des fausses nouvelles. Voici quelques-unes des études notables :

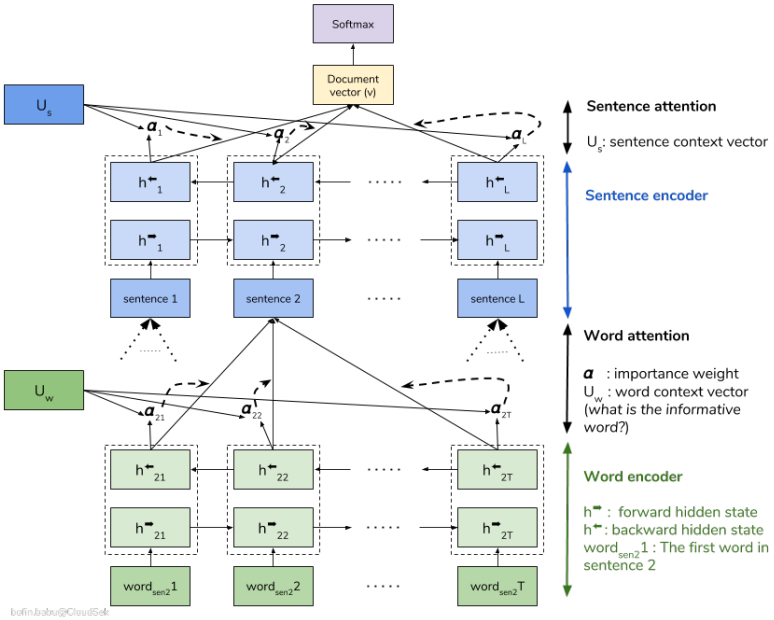


FIGURE 2.14 – Modèle HAN [6]

Le Gated Recurrent Unit (GRU) incorpore des unités de porte qui régulent le flux d'information à l'intérieur de l'unité, similaire à une unité LSTM. Néanmoins, le GRU possède moins de paramètres puisqu'il n'a pas de porte de sortie, ce qui entraîne une structure plus simple comparée au processus LSTM [178]. La formulation fondamentale d'une couche GRU est la suivante :

$$z_t = \sigma(x_t U_z + h_{t-1} W_z) \tag{2.14}$$

$$r_t = \sigma(x_t U_r + h_{t-1} W_r) \tag{2.15}$$

$$\tilde{h}_t = \tanh(x_t U_h + (h_{t-1} \odot r_t) W_h) \tag{2.16}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{2.17}$$

La porte de réinitialisation r_t détermine comment la nouvelle entrée collabore avec la mémoire précédente, tandis que la porte de mise à jour z_t spécifie dans quelle mesure la mémoire précédente est intégrée dans l'étape temporelle actuelle. De plus, le terme $\tilde{h}_t$ représente l'activation candidate de l'état caché h_t.

Yang et al. [6] ont proposé un modèle basé sur HAN qui combine le réseau d'attention hiérarchique au niveau des mots et des phrases pour capturer la structure hiérarchique des articles de presse. Le modèle a bien performé sur plusieurs jeux de données de référence.

Li et al.[179] ont proposé une approche basée sur HAN pour la détection des fausses nouvelles qui utilise un mécanisme de fusion de caractéristiques novateur. Ils ont utilisé un ensemble de données d'articles de presse étiquetés et ont atteint une précision de 95,7%.

Dans une étude plus récente publiée dans IEEE Access, Wang et al.[95] ont proposé un modèle basé sur HAN qui utilise à la fois des caractéristiques textuelles et visuelles pour détecter les fausses nouvelles sur les réseaux sociaux. Les auteurs ont incorporé un réseau de neurones convolutif (CNN) pour extraire des caractéristiques visuelles à partir des images, et un HAN pour capturer la structure hiérarchique du contenu textuel. Le modèle proposé a atteint une précision supérieure à plusieurs autres modèles de référence.

Une autre étude publiée dans IEEE Access par Li et al.[180] a proposé un modèle basé sur HAN qui utilise l'apprentissage multi-tâches pour détecter simultanément les fausses nouvelles et distinguer les différents types de fausses nouvelles. Le modèle a bien performé sur plusieurs jeux de données, démontrant l'efficacité de l'approche proposée. Dans une étude publiée dans le Journal of Ambient Intelligence and Humanized Computing, Dai et al.[181] ont développé un cadre basé sur HAN qui intègre des graphes de connaissances externes pour détecter les fausses nouvelles dans les articles de presse. Le modèle a atteint une performance de pointe sur plusieurs jeux de données de référence.

Kaur et al.[182] ont proposé un modèle HAN qui incorpore à la fois des caractéristiques textuelles et visuelles pour détecter les fausses nouvelles dans les images. Le modèle a atteint une haute précision dans la détection des images de fausses nouvelles sur l'ensemble de données Snopes.

D'autres études ont exploré l'utilisation de HANs en combinaison avec d'autres techniques, telles que les réseaux convolutionnels de graphes (GCNs) et l'entraînement adversarial. Par exemple, Chen et al.[183] ont proposé un modèle HAN-GCN qui intègre des informations graphiques pour obtenir de meilleures performances dans la détection des fausses nouvelles. Dans une étude de Chen et al.[183], un modèle HAN avec un entraînement adversarial a montré une amélioration de la robustesse de la détection des fausses nouvelles.

D'autres études ont exploré l'utilisation des HANs spécifiquement pour identifier les caractéristiques linguistiques des fausses nouvelles. Par exemple, dans une étude publiée dans le journal Applied Sciences, Gao et al.[181] ont développé un modèle de détection des fausses nouvelles qui utilise des HANs pour analyser la complexité linguistique des articles de presse. Le modèle a atteint une précision de 87,3% sur un ensemble de données d'articles de presse réels et faux.

Enfin, dans une étude publiée dans Information Processing and Management, Zhou et al.[184] ont proposé un modèle basé sur HAN qui exploite des sources de connaissances externes pour améliorer la détection des fausses nouvelles. Les auteurs ont incorporé les connaissances provenant de sources externes telles que les embeddings de mots et les structures de réseaux sociaux dans le modèle HAN. Le modèle proposé a surpassé plusieurs autres méthodes de référence sur deux jeux de données de référence.

De même, certaines études ont exploré l'utilisation des HANs en échange d'autres types de données, comme les données des réseaux sociaux ou les commentaires des utilisateurs. Par exemple, dans une étude publiée dans le journal Social Network Analysis and Mining, Hu et al.[185] ont développé un modèle de détection des fausses nouvelles qui combine des HANs avec des techniques d'analyse de réseaux sociaux. Le modèle a atteint une précision de 83,9% sur un ensemble de données d'articles de fausses nouvelles et de commentaires d'utilisateurs.

En résumé, les HANs ont montré un grand potentiel dans la détection des fausses nouvelles et de la propagande, en particulier lorsqu'ils sont combinés avec d'autres techniques d'apprentissage automatique et des sources de données multimodales. Les études mentionnées ci-dessus fournissent des informations importantes sur l'utilisation des HANs pour la détection des fausses nouvelles et offrent des directions prometteuses pour la recherche future dans ce domaine.

Algorithm 8 Algorithme de Réseau Hiérarchique à Attention pour la Détection des Fausses Nouvelles

function TRAINHANMODEL(training_data, labels, num_epochs, batch_size, learning_rate, embedding_dim, vocab_size, max_seq_length, lstm_units, attention_units, dropout_rate)
    Initialize Hierarchical Attention Network with specified parameters
    for each epoch from 1 to num_epochs **do**
        for each batch in training_data divided by batch_size **do**
            $X\_batch, y\_batch \leftarrow$ GetNextBatch(training_data, labels, batch_size)
            predictions $\leftarrow$ ForwardPass(HAN, X_batch)
            loss $\leftarrow$ ComputeLoss(predictions, y_batch)
            gradients $\leftarrow$ BackwardPass(loss)
            UpdateWeights(HAN, gradients, learning_rate)
    return Trained Hierarchical Attention Network
function PREDICT(HAN_model, test_data)
    Initialize predictions to an empty list
    for each x in test_data **do**
        prediction $\leftarrow$ ForwardPass(HAN_model, x)
        Add prediction to predictions
    return predictions
function FORWARDPASS(HAN_model, X)
    embeddings $\leftarrow$ Embed(X, vocab_size, embedding_dim)
    word_level_lstm_output $\leftarrow$ BI-LSTM(embeddings, lstm_units)
    word_level_attention_output $\leftarrow$ Attention(word_level_lstm_output, attention_units)
    sentence_level_lstm_output $\leftarrow$ BI-LSTM(word_level_attention_output, lstm_units)
    sentence_level_attention_output $\leftarrow$ Attention(sentence_level_lstm_output, attention_units)
    dropout_output $\leftarrow$ Dropout(sentence_level_attention_output, dropout_rate)
    final_output $\leftarrow$ Dense(dropout_output, num_classes, activation='softmax')
    return final_output
function COMPUTELOSS(predictions, labels)
    loss $\leftarrow$ CategoricalCrossentropy(predictions, labels)
    return loss
function BACKWARDPASS(loss)
    gradients $\leftarrow$ ComputeGradients(loss)
    return gradients
function UPDATEWEIGHTS(HAN_model, gradients, learning_rate)
    ApplyGradients(HAN_model, gradients, learning_rate)
    return
function EMBED(X, vocab_size, embedding_dim)
    embeddings $\leftarrow$ EmbeddingLayer(X, vocab_size, embedding_dim)
    return embeddings
function ATTENTION(lstm_output, attention_units)
    attention_weights $\leftarrow$ Dense(lstm_output, attention_units, activation='tanh')
    context_vector $\leftarrow$ DotProduct(attention_weights, lstm_output)
    return context_vector

2.7 Fonctions d'activation

Les fonctions d'activation jouent un rôle crucial dans les réseaux de neurones artificiels, influençant directement les performances du modèle et la manière dont les informations sont propagées et transformées à travers les couches d'un réseau. Leur objectif principal est de décider si un neurone particulier doit être activé ou non en appliquant une transformation non linéaire aux entrées reçues, permettant ainsi au modèle d'apprendre et de résoudre des problèmes complexes. Cette section présente les principales fonctions d'activation, leurs avantages, leurs inconvénients et leurs domaines d'utilisation.

2.7.1 Fonction Sigmoïde (Sigmoid)

La fonction Sigmoïde, également connue sous le nom de fonction logistique[186], est définie mathématiquement comme suit :

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{2.18}$$

où x est l'entrée, et $\sigma(x)$ est la sortie de la fonction. La fonction Sigmoïde prend une valeur d'entrée réelle et la transforme en une sortie dans l'intervalle (0, 1), ce qui est particulièrement utile pour les tâches de classification binaire. La sortie peut être interprétée comme une probabilité, avec des valeurs proches de 0 indiquant une forte probabilité que l'entrée appartienne à une classe, et des valeurs proches de 1 indiquant une forte probabilité pour l'autre classe[187].

Cette fonction est largement utilisée dans les premières architectures de réseaux de neurones, notamment dans les réseaux neuronaux multicouches [188, 189] et les réseaux de neurones récurrents. Cependant, elle a été progressivement remplacée par d'autres fonctions d'activation plus performantes dans les réseaux profonds.

Avantages et inconvénients : La fonction Sigmoïde présente plusieurs avantages dans certains contextes, mais elle a aussi des limitations importantes.

Tout d'abord, un de ses principaux avantages réside dans sa capacité à générer des sorties lisses et différentiables dans un intervalle de probabilités, ce qui est particulièrement adapté pour les tâches de classification binaire et pour la rétropropagation du gradient (backpropagation). Cette propriété permet une optimisation efficace grâce à des méthodes de gradient, ce qui a été l'une des raisons pour lesquelles cette fonction a été largement adoptée dans les premières architectures de réseaux neuronaux .

Cependant, la fonction Sigmoïde souffre de plusieurs inconvénients majeurs, notamment le problème de *gradient qui disparaît* (*vanishing gradient problem*). Lorsque la valeur d'entrée est soit très élevée (positive) soit très basse (négative), la dérivée de la fonction Sigmoïde devient extrêmement petite. Par conséquent, les gradients associés à ces neurones peuvent devenir presque nuls lors de la rétropropagation, rendant l'apprentissage extrêmement lent pour les couches profondes du réseau. Ce problème est particulièrement prévalent dans les réseaux neuronaux profonds, où plusieurs couches utilisent cette fonction, ce qui ralentit considérablement la convergence des poids .

Un autre inconvénient est que la fonction Sigmoïde n'est pas centrée autour de zéro. Les valeurs de sortie sont toujours positives, ce qui peut provoquer des déséquilibres dans les poids mis à jour. Lorsque les poids sont ajustés, cela peut entraîner une instabilité de l'apprentissage, car les gradients sont toujours positifs, forçant le modèle à faire des ajustements moins précis. Par conséquent, des fonctions comme Tanh, qui sont centrées autour de zéro, sont souvent préférées dans des scénarios similaires[190, 191].

Enfin, la fonction Sigmoïde est sujette à la saturation, c'est-à-dire que pour des valeurs très élevées ou très basses, la sortie est respectivement proche de 1 ou de 0, entraînant une perte de précision dans les gradients et l'apprentissage. Cela peut aussi conduire à un ralentissement de la convergence du réseau, nécessitant plus d'itérations d'apprentissage .

Domaine d'utilisation : La fonction Sigmoïde est principalement utilisée dans les tâches de classification binaire, ainsi que dans les réseaux neuronaux récurrents (RNN), où elle peut être utile pour maintenir des activations dans une plage limitée. Bien qu'elle soit moins utilisée dans les réseaux neuronaux profonds aujourd'hui, elle reste pertinente pour les modèles moins complexes et les problèmes de classification binaires simples .

2.7.2 Fonction Tangente Hyperbolique (Tanh)

La fonction Tangente Hyperbolique, ou *Tanh*, est une fonction d'activation non linéaire couramment utilisée dans les réseaux de neurones [189], en particulier dans les réseaux de neurones récurrents (RNN). Elle est définie par la formule suivante :

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.19}$$

La fonction Tanh est une fonction sigmoïde modifiée qui convertit une entrée réelle en une sortie comprise entre -1 et 1, contrairement à la fonction sigmoïde classique qui est bornée entre 0 et 1. Cette caractéristique rend Tanh plus adaptée dans de nombreux cas, notamment dans les réseaux où les données sont centrées autour de zéro[188].

Avantages : La fonction Tanh présente plusieurs avantages qui la rendent plus efficace que la fonction Sigmoïde dans certains scénarios :

— Centrée sur zéro : Contrairement à la fonction sigmoïde, dont les sorties sont limitées entre 0 et 1, la fonction Tanh produit des sorties entre -1 et 1. Cela signifie que les valeurs négatives peuvent être prises en compte dans les prédictions, ce qui facilite l'apprentissage dans des réseaux où les données sont symétriques autour de zéro. Cela permet de mieux équilibrer les poids dans les couches du réseau de neurones pendant la phase d'apprentissage[187, 192].

— Performance sur les données normalisées : La fonction Tanh est souvent utilisée lorsque les données sont centrées autour de zéro après une normalisation. Elle aide à capturer des dépendances complexes dans les données, ce qui la rend efficace pour des tâches non linéaires complexes comme dans les réseaux de neurones récurrents ou les réseaux neuronaux convolutionnels profonds[193].

— Saturation moins sévère : Bien que la fonction Tanh puisse aussi saturer (comme la Sigmoïde), elle a tendance à saturer moins sévèrement pour des valeurs modérées d'entrée[194]. Cela peut conduire à des gradients légèrement plus stables lors de la rétropropagation, par rapport à la fonction Sigmoïde .

Inconvénients : Malgré ses avantages, la fonction Tanh présente certains inconvénients qui peuvent limiter son efficacité dans certaines architectures de réseaux de neurones :

— Problème de gradient qui disparaît : Comme la fonction Sigmoïde, la fonction Tanh souffre également du problème de *vanishing gradient*, surtout pour des valeurs d'entrée extrêmes. Lorsque les entrées deviennent trop grandes ou trop petites, la fonction Tanh commence à saturer, ce qui conduit à des gradients proches de zéro. Cela ralentit considérablement l'apprentissage, en particulier dans les réseaux profonds avec plusieurs couches [195].

— Coût computationnel élevé : La fonction Tanh implique le calcul de fonctions exponentielles, ce qui la rend plus coûteuse en termes de calcul que des fonctions d'activation plus simples comme ReLU. Ce coût supplémentaire peut devenir significatif dans les architectures de réseaux neuronaux très profonds ou pour des tâches nécessitant un traitement en temps réel .

— Moins adaptée aux couches profondes : Bien que la fonction Tanh soit efficace dans les couches de surface des réseaux neuronaux, elle est moins performante dans les réseaux neuronaux très profonds. Pour ces réseaux, des fonctions comme ReLU et ses variantes sont préférées, car elles atténuent le problème du gradient qui disparaît et offrent de meilleures performances globales sur les grandes architectures neuronales .

La fonction Tanh est un choix populaire pour les réseaux de neurones lorsqu'il s'agit de modéliser des données symétriques et non linéaires. Sa capacité à produire des sorties centrées autour de zéro la rend particulièrement utile dans les réseaux récurrents (RNN). Cependant, elle souffre encore du problème de gradient qui disparaît, ce qui limite son efficacité dans les réseaux profonds. Néanmoins, elle reste un bon compromis entre simplicité et capacité de modélisation non linéaire.

2.7.3 Fonction Rectified Linear Unit (ReLU)

La fonction ReLU est l'une des fonctions d'activation les plus populaires dans les réseaux de neurones profonds. Elle est définie comme suit :

$$\mathrm{ReLU}(x) = \max(0, x) \tag{2.20}$$

La fonction ReLU renvoie la valeur d'entrée lorsqu'elle est positive, et 0 sinon. Elle introduit une non-linéarité tout en restant simple à calculer.

Domaine d'application ReLU est principalement utilisée dans les réseaux de neurones convolutionnels, les réseaux profonds, et d'autres architectures modernes de deep learning. Elle est particulièrement

efficace pour traiter des problèmes liés à l'image, au texte et à d'autres formes de données non structurées [188].

Avantages La fonction ReLU a gagné en popularité pour plusieurs raisons. Contrairement aux fonctions d'activation sigmoïde et tangente hyperbolique, ReLU n'a pas de problème de saturation dans les valeurs positives. En conséquence, elle permet une rétropropagation plus efficace des gradients, ce qui accélère l'apprentissage dans les réseaux de neurones profonds [196]. ReLU est également rapide à calculer, car elle ne nécessite qu'une simple comparaison, ce qui améliore la vitesse d'exécution globale du réseau . Cette efficacité est particulièrement utile pour les réseaux profonds avec des millions de paramètres.

Un autre avantage majeur est que ReLU aide à atténuer le problème du gradient qui disparaît, souvent observé dans les couches profondes des réseaux de neurones lorsqu'on utilise des fonctions comme Sigmoid ou Tanh .

Inconvénients Malgré ses avantages, ReLU présente certains inconvénients notables. L'un des plus importants est le problème des *neurones morts*. Si un neurone reçoit constamment des entrées négatives, la sortie de ReLU sera toujours zéro, empêchant le neurone de contribuer à l'apprentissage. Ce problème peut entraîner la mort de certains neurones, qui ne s'activent plus pendant tout le processus d'entraînement. Cette difficulté peut être partiellement atténuée en utilisant des variantes comme Leaky ReLU ou Parametric ReLU [195].

De plus, bien que ReLU soit efficace dans les couches intermédiaires d'un réseau, son comportement non borné pour les valeurs positives peut conduire à des activations trop importantes. Cela peut, dans certains cas, déstabiliser le réseau pendant l'entraînement et nécessiter l'utilisation de techniques comme la normalisation par lots (*batch normalization*) pour stabiliser les gradients.

2.7.4 Fonction Leaky ReLU

La fonction Leaky ReLU est une variante de la fonction ReLU, qui vise à résoudre le problème des « neurones morts » (Dead Neurons) rencontré dans ReLU. Ce problème survient lorsque les valeurs négatives d'entrée conduisent à des gradients nuls, empêchant ainsi ces neurones de contribuer à l'apprentissage. Leaky ReLU introduit une petite pente pour les valeurs négatives, définie par :

$$\text{Leaky ReLU}(x) = \begin{cases} x & \text{si } x > 0 \\ \alpha x & \text{si } x \leq 0 \end{cases} \tag{2.21}$$

où α est un petit facteur, souvent fixé à une valeur comme 0.01, permettant un flux de gradient même pour des valeurs négatives .

Avantages et inconvénients La fonction Leaky ReLU présente plusieurs avantages par rapport à la ReLU classique. Tout d'abord, elle résout partiellement le problème des neurones morts en introduisant

un petit gradient dans la région négative de l'entrée, ce qui permet aux neurones d'apprendre même si leur sortie est négative. Cela aide à éviter des situations où une grande partie du réseau devient inactive pendant l'entraînement. Contrairement à la fonction ReLU qui bloque complètement les valeurs négatives en renvoyant zéro, Leaky ReLU offre une meilleure capacité d'apprentissage dans les réseaux profonds en facilitant la propagation des gradients.

Cependant, Leaky ReLU n'est pas exempte de limitations. Le principal inconvénient est que le choix de la valeur de α est un hyperparamètre sensible qui peut nécessiter des ajustements manuels ou une validation croisée pour chaque jeu de données ou modèle spécifique. Une valeur inappropriée de α pourrait entraîner une convergence plus lente ou des résultats sous-optimaux [195]. De plus, comme avec ReLU, la fonction Leaky ReLU n'est pas centrée sur zéro, ce qui peut provoquer un certain biais dans la mise à jour des poids pendant l'entraînement [187].

2.7.5 Fonction Softmax

Les réseaux de neurones utilisent fréquemment la fonction Softmax pour activer les couches de sortie des modèles de classification multi-classes. Elle est définie comme suit :

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{2.22}$$

où z_i est la valeur de la i-ème unité de sortie, et la somme dans le dénominateur est effectuée sur toutes les unités de sortie j. La fonction Softmax transforme les valeurs d'entrée en probabilités normalisées, ce qui signifie que la sortie de la fonction Softmax est un vecteur dont la somme des composantes est égale à 1. Cette propriété en fait une fonction idéale pour les problèmes de classification où chaque entrée doit être classée en une seule classe parmi plusieurs.

Domaine d'utilisation Dans les couches de sortie des réseaux de neurones, la fonction Softmax est principalement employée pour des tâches de classification multi-classes. Elle est efficace pour convertir les scores bruts (logits) en probabilités interprétables, permettant ainsi de faire des prévisions sur la classe à laquelle un échantillon appartient. En raison de sa capacité à fournir une distribution de probabilité, elle est particulièrement utile dans des domaines tels que la reconnaissance d'images, la classification de texte, et la reconnaissance vocale.

Avantages et Inconvénients La fonction Softmax présente plusieurs avantages significatifs. Tout d'abord, elle fournit une sortie qui est directement interprétable comme une probabilité, facilitant ainsi la prise de décision. Les sorties sont normalisées pour être dans la plage [0,1], ce qui simplifie l'interprétation des résultats. De plus, la Softmax est différentiable, ce qui permet l'utilisation de la rétropropagation pour l'entraînement des modèles.

Cependant, la fonction Softmax présente également des inconvénients. L'un des principaux problèmes est la sensibilité aux valeurs extrêmes. Les grandes valeurs des logits peuvent entraîner des probabilités proches de 1 pour certaines classes et des probabilités proches de 0 pour les autres, ce qui peut rendre le modèle sur-confiant et moins robuste face aux variations des données. De plus, la Softmax est sensible aux déséquilibres dans les classes ; lorsque les données sont fortement déséquilibrées, la fonction peut amplifier les biais existants dans les données d'entraînement. Enfin, le calcul de la Softmax peut être numériquement instable lorsque les valeurs d'entrée sont très grandes ou très petites, ce qui peut entraîner des problèmes de précision.

2.7.6 Fonction Swish

La fonction d'activation Swish, introduite par Ramachandran et al. en 2017, est définie par la formule suivante :

$$\text{Swish}(x) = x \cdot \sigma(x) \tag{2.23}$$

où $\sigma(x)$ représente la fonction sigmoïde $\sigma(x) = \frac{1}{1+e^{-x}}$. Cette fonction d'activation a été proposée comme une alternative aux fonctions d'activation plus classiques telles que ReLU et ses variantes.

Domaine d'Utilisation : La fonction Swish est essentiellement employée dans les réseaux de neurones profonds, tels que les architectures de réseaux convolutifs et les réseaux de neurones récurrents. Grâce à sa souplesse et à ses caractéristiques uniques, il est intéressant de le choisir pour diverses applications, comme la vision par ordinateur et le traitement du langage naturel.

La fonction Swish présente plusieurs avantages notables :

— **Non-Linéarité Améliorée :** Contrairement à ReLU, qui est non-linéaire uniquement pour les valeurs positives, Swish est non-linéaire sur tout le domaine. Cette propriété permet aux modèles d'apprendre des représentations plus complexes et non-linéaires des données.

— **Gradient Non-Nul :** Swish offre un gradient non-nul pour toutes les valeurs d'entrée, ce qui peut aider à atténuer les problèmes de gradient nul rencontrés avec ReLU pour les valeurs négatives. Cela peut faciliter l'entraînement des réseaux profonds en réduisant les risques d'explosion ou de disparition du gradient.

— **Amélioration des Performances :** Des recherches ont montré que Swish peut améliorer les performances de certains modèles par rapport à ReLU, en particulier pour des architectures profondes. Par exemple, la recherche de Ramachandran et al. (2017) a démontré que Swish pouvait surpasser ReLU sur plusieurs benchmarks de vision par ordinateur.

Cependant, la fonction Swish n'est pas exempte de limitations :

— **Complexité Computationnelle :** Swish est plus coûteuse à calculer que ReLU en raison de la fonction sigmoïde impliquée. Cette complexité accrue peut entraîner des temps d'entraînement plus longs et une utilisation plus importante des ressources computationnelles.

— **Difficulté de Réglage des Hyperparamètres :** Comme pour d'autres fonctions d'activation non linéaires, l'optimisation des hyperparamètres peut être plus complexe. Il peut être nécessaire de tester plusieurs configurations pour obtenir les meilleurs résultats.

— **Moins de Support dans les Frameworks :** Bien que le support pour Swish soit en augmentation, elle peut ne pas être aussi largement supportée dans tous les frameworks et bibliothèques d'apprentissage automatique comparé à ReLU ou ses variantes.

2.7.7 Étude comparative

La table 2.3 compare les principales fonctions d'activation selon différents critères :

TABLE 2.3 – Comparaison des fonctions d'activation

Fonction	Plage de sortie	Centrée sur zéro	Saturation	Problème de gradient	Cas d'utilisation
Sigmoïde	(0, 1)	Non	Oui	Oui	Classification binaire
Tanh	(-1, 1)	Oui	Oui	Oui	RNN, modélisation non linéaire
ReLU	(0, ∞)	Non	Non	Neurones morts	CNN, réseaux profonds
Leaky ReLU	(-∞, ∞)	Non	Non	Moins de neurones morts	Réseaux profonds
Softmax	(0, 1)	Non	Oui	Oui	Classification multi-classes
Swish	(-∞, ∞)	Non	Faible	Non	Architectures complexes

Le choix de la fonction d'activation dépend fortement de la nature du problème à résoudre. Les fonctions comme Sigmoïde et Softmax sont utiles pour les tâches de classification, tandis que ReLU et ses variantes sont préférées dans les réseaux profonds en raison de leur capacité à résoudre le problème du gradient qui disparaît. Des fonctions plus récentes comme Swish montrent des performances prometteuses, en particulier dans des architectures complexes.

Conclusion

La détection de fausses informations basée sur le texte est un domaine de recherche dynamique et essentiel pour lutter contre la désinformation en ligne. Ce chapitre a mis en lumière les différentes étapes du processus, depuis le prétraitement des données textuelles jusqu'à l'application d'algorithmes de classification sophistiqués. Nous avons discuté des méthodes de représentation textuelle, telles que le sac de mots et les plongements de mots, qui transforment les données textuelles en formats exploitables par les machines. Ensuite, nous avons exploré divers algorithmes de classification, y compris les modèles d'apprentissage automatique classiques, les modèles ensemblistes et les modèles d'apprentissage profond. Chaque méthode présente ses propres avantages et inconvénients, mais l'usage combiné de plusieurs techniques permet souvent d'améliorer les performances globales du système.Les progrès récents dans le domaine de l'apprentissage profond, notamment avec l'émergence des réseaux de neurones récurrents et des architectures d'attention hiérarchique, ouvrent des perspectives encourageantes pour perfectionner

l'identification des informations trompeuses ou erronées. En conclusion, bien que des défis persistent, les progrès continus dans ce domaine offrent des perspectives encourageantes pour développer des systèmes de détection de fausses informations de plus en plus efficaces et robustes.

Chapitre 3

Ensembles de Données, Méthodes d'Optimisation et Analyse Comparative en Détection des Fausses Nouvelles

Introduction

Dans le chapitre précédent, intitulé Processus de détection de fausses informations textuelles, nous avons exploré les différentes étapes du traitement des données textuelles. Nous avons présenté le prétraitement des textes, les méthodes de représentation textuelle, ainsi que les algorithmes de classification utilisés dans les approches d'apprentissage automatique et d'apprentissage profond.

Le présent chapitre poursuit cette analyse en se concentrant sur les ensembles de données, les techniques d'optimisation, et les méthodes d'évaluation utilisées pour mesurer la performance des modèles de détection. Nous introduirons tout d'abord les principaux ensembles de données utilisés dans cette étude, ensuite, nous détaillerons les différentes techniques d'optimisation employées. Une analyse comparative de ces optimiseurs sera également présentée. Nous passerons ensuite aux métriques d'évaluation qui jouent un rôle clé dans la validation de la précision des modèles. Enfin, ce chapitre propose une étude comparative approfondie entre plusieurs architectures de réseaux neuronaux. Une évaluation expérimentale sera réalisée pour analyser leurs performances respectives.

3.1 Ensembles de Données

L'analyse comparative s'appuie sur quatre corpus de données étalons dédiés à l'identification de la désinformation. Ces corpus englobent divers domaines comme la sphère politique, le secteur de la santé, le monde des affaires et l'univers technologique. Cette approche vise à minimiser les distorsions dans la recherche et à permettre des déductions pertinentes. La figure 4.2 présente les nuages de mots de chaque ensemble de données. De plus, des ensembles de données de détection de fausses nouvelles binaires et multi-classes sont représentés dans cet ensemble. Nous avons choisi ces ensembles de données en nous basant sur des recherches antérieures sur la détection de fausses nouvelles, puisque cette étude se

concentre sur la détection de fausses nouvelles basée sur le texte. Les principales caractéristiques de chaque ensemble de données sont présentées dans le tableau 3.2.

3.1.1 Liar Dataset

Ce jeu de données comprend 12,8 K de déclarations courtes étiquetées manuellement provenant du site de vérification des faits Politifact.com[197]. Chaque déclaration est évaluée par un éditeur de Politifact.com [197] pour sa véracité. Le jeu de données possède six étiquettes détaillées : pants-fire (mensonge flagrant), false (faux), barely-true (à peine vrai), half-true (moitié vrai), mostly-true (principalement vrai) et true (vrai). La distribution des étiquettes est relativement bien équilibrée. Pour nos besoins, les six étiquettes détaillées du jeu de données ont été regroupées en une classification binaire, c'est-à-dire l'étiquette 1 pour les fausses nouvelles et l'étiquette 0 pour les nouvelles fiables. Ce choix a été fait en raison de la fonctionnalité binaire du jeu de données Fake News. Le jeu de données est divisé en trois fichiers : 1) Ensemble d'entraînement : 5770 vraies nouvelles et 4497 fausses nouvelles ; 2) Ensemble de test : 1382 vraies nouvelles et 1169 fausses nouvelles ; 3) Ensemble de validation : 1382 vraies nouvelles et 1169 fausses nouvelles. Les trois sous-ensembles sont bien équilibrés, il n'est donc pas nécessaire de procéder à un suréchantillonnage ou à un sous-échantillonnage.

Le jeu de données[31] traité a été téléchargé sur Google Drive puis chargé dans Jupyter de Colab sous forme de DataFrame Pandas. Une nouvelle colonne contenant le nombre de mots pour chaque article a été ajoutée.L'application de la fonction df.describe() à cette colonne permet d'afficher un résumé statistique comprenant : le nombre total d'entrées (15389), la moyenne (17,96), l'écart-type (8,57), la valeur minimale (1), le premier quartile (12), la médiane (17), le troisième quartile (22), et la valeur maximale (66).Ces statistiques montrent qu'il y a des articles avec un seul mot dans le jeu de données, il a donc été décidé de supprimer toutes les lignes avec moins de 10 mots car elles sont considérées comme peu informatives. Le jeu de données résultant contient 1657 lignes de moins que l'original.L'analyse statistique mise à jour de cette colonne révèle les informations suivantes : un total de 13 732 entrées, une moyenne de 19,23, un écart-type de 8,19, une valeur minimale de 10, un premier quartile de 14, une médiane de 18, un troisième quartile de 23, et une valeur maximale de 66. En conclusion, la longueur moyenne des articles est de 19 mots.

3.1.2 FakeNewsNet

Ce jeu de données a été construit en rassemblant des informations provenant de deux sites de vérification des faits pour obtenir du contenu de nouvelles fausses et vraies, comme PolitiFact et GossipCop. Chez PolitiFact, des journalistes et des experts du domaine examinent les nouvelles politiques et fournissent des résultats d'évaluation de vérification des faits pour qualifier les articles de presse comme étant faux ou vrais. En revanche, chez GossipCop, les histoires de divertissement provenant de divers médias sont évaluées par un score de notation sur une échelle de 0 à 10, indiquant le degré de fausseté à véracité. Le jeu de données contient environ 900 nouvelles politiques et 20k nouvelles de potins et n'a

que deux étiquettes : vrai et faux. Ce jeu de données est disponible publiquement grâce aux fonctions fournies par l'équipe de FakeNewsNet et l'API Twitter. Comme mentionné ci-dessus, FakeNewsNet peut être divisé en deux sous-ensembles : GossipCop et Politifact.com. Nous avons décidé d'analyser uniquement les nouvelles politiques car elles produisent des conséquences plus graves dans le monde réel que les potins. Le jeu de données est bien équilibré et contient 434 vraies nouvelles et 367 fausses nouvelles. La plupart des nouvelles concernent les États-Unis, comme cela a déjà été noté dans LIAR. Les sujets des fausses nouvelles concernent Obama, la police, Clinton et Trump, tandis que les sujets des vraies nouvelles concernent Trump, les Républicains et Obama.À l'instar du jeu de données LIAR, une colonne supplémentaire a été intégrée. L'utilisation de la commande df.describe() sur cette nouvelle colonne a généré le résumé statistique suivant :

— Nombre total d'entrées : 801

— Moyenne : 1459,22

— Écart-type : 3141,16

— Valeur minimale : 3

— Premier quartile (25%) : 114

— Médiane (50%) : 351

— Troisième quartile (75%) : 893

— Valeur maximale : 17377

En comparant les deux jeux de données, on constate une différence significative dans la longueur des articles. Le jeu de données Politifact présente une moyenne de 1459 mots par article, ce qui est considérablement plus élevé que la moyenne de 19 mots par article observée dans le jeu de données LIAR. Ces statistiques nous ont suggéré de comparer les performances du modèle sur des jeux de données avec des caractéristiques si différentes.

3.1.3 ISOT Fake News

Le jeu de données ISOT Fake News [93, 36] est un ensemble de données open-source développé pour aborder le problème des fausses nouvelles et de la désinformation dans le paysage médiatique. Ce jeu de données est principalement composé de contenu textuel, tel que des articles de presse et des titres, et il sert de ressource essentielle pour le développement d'algorithmes capables de détecter les informations trompeuses ou mensongères.

Caractéristiques du jeu de données ISOT Fake News Origine et Authenticité : Le jeu de données ISOT a été collecté et vérifié à partir de sources fiables en utilisant des plateformes établies d'authentification des fausses nouvelles et des sites de vérification des faits comme Snopes.com. Il comprend des nouvelles provenant de divers domaines pour introduire une diversité et capturer la complexité du contenu en ligne.

Structure et Composition :

Étiquetage binaire : Chaque élément du jeu de données est catégorisé de manière binaire, soit comme "faux", soit comme "vrai". Cette classification dichotomique est particulièrement adaptée pour :

— L'entraînement de modèles d'apprentissage automatique.

— L'évaluation de la performance de ces modèles.

Segmentation du jeu de données : Les données sont réparties en plusieurs ensembles distincts :

— Un ensemble d'entraînement

— Un ensemble de test

— Un ensemble de validation

Cette division tripartite du jeu de données permet :

— Un processus d'apprentissage contrôlé pour les modèles

— Une évaluation précise et fiable de leurs performances

— Une validation robuste des résultats obtenus

Cette structure bien pensée facilite le développement, l'ajustement et la validation de modèles d'apprentissage automatique dédiés à la détection de fausses informations.

Utilisation et Applications : Le jeu de données ISOT est utilisé pour développer des algorithmes de détection des fausses nouvelles basés sur le texte. Il permet aux chercheurs d'analyser les schémas de diffusion de la désinformation et joue un rôle significatif dans les efforts mondiaux visant à améliorer la littératie médiatique et à garantir le partage d'informations précises.

Pour garantir la qualité et l'informativité des données, les articles contenant moins de 10 mots ont été supprimés, car ils sont considérés comme peu informatifs. Les statistiques suivantes ont été calculées pour le jeu de données traité :

— Nombre d'articles : 13 732

— Nombre moyen de mots par article : 19.23

— Écart-type du nombre de mots : 8.19

— Nombre minimum de mots par article : 10

— Nombre maximum de mots par article : 66

Ces statistiques permettent de mieux comprendre la distribution des longueurs d'articles dans le jeu de données et d'ajuster les modèles en conséquence.

Importance et Contribution : Le jeu de données ISOT Fake News est un outil crucial pour les chercheurs travaillant sur la détection des fausses nouvelles. En offrant une base bien étiquetée et diversifiée, il permet le développement et l'évaluation de modèles sophistiqués capables de distinguer les informations vraies des informations fausses. Cela contribue non seulement à la recherche académique mais

aussi à des applications pratiques visant à améliorer la qualité des informations disponibles en ligne et à lutter contre la propagation de la désinformation.

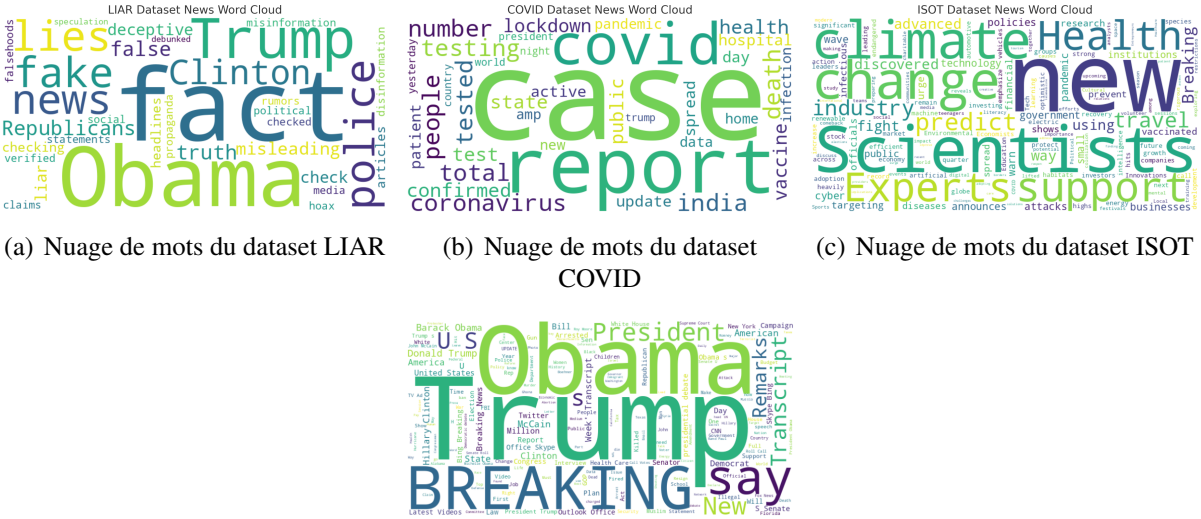
Ce jeu de données continue de servir de référence pour les nouvelles méthodologies et les techniques innovantes dans le domaine de la détection des fausses nouvelles, aidant ainsi à protéger la société contre les impacts négatifs de la désinformation.

3.1.4 COVID dataset

Le jeu de données COVID [198] a été rassemblé dans le but de détecter les fausses nouvelles liées à la pandémie de COVID-19. Les articles ont été extraits de Twitter et vérifiés par des sites faisant autorité tels que politifact.com et snopes.com. Composé de 10 700 entrées rédigées en anglais, le jeu de données se divise en deux catégories : les vraies nouvelles et les fausses nouvelles.

• Réelles : Tweets provenant de sources vérifiées et donnant des informations utiles sur le COVID-19.

• Fausses : Tweets, publications, articles faisant des affirmations et des spéculations sur le COVID-19, vérifiés comme étant faux.



(a) Nuage de mots du dataset LIAR (b) Nuage de mots du dataset COVID (c) Nuage de mots du dataset ISOT

(d) Nuage de mots du dataset FakeNewsNet

FIGURE 3.1 – Nuages de mots de chaque jeu de données utilisé dans les études

TABLE 3.1 – Caractéristiques principales des ensembles de données de référence utilisés dans nos études.

Jeu de données	Domaine	Média	Vérification des faits	Taille	Nombre de classes
Liar [31]	Politique	Médias traditionnels	Rédacteurs & journalistes	12 836	6
ISOT [93, 36]	Politique & Actualités mondiales	Médias traditionnels	Sites de vérification des faits	44 866	2
Covid [198]	Covid-19 & Santé	Twitter	Sites de vérification des faits	10 700	2

TABLE 3.2 – Caractéristiques principales des ensembles de données de référence utilisés dans nos études.

Jeu de données	**Domaine**	**Média**	**Vérification des faits**	**Taille**	**Nombre de classes**
Liar [31]	Politique	Médias traditionnels	Rédacteurs & journalistes	12 836	6
ISOT [93, 36]	Politique & Actualités mondiales	Médias traditionnels	Sites de vérification des faits	44 866	2
Covid [198]	Covid-19 & Santé	Twitter	Sites de vérification des faits	10 700	2
FakeNewsNet [85]	Politique & Divertissement	Twitter Facebook	Sites de vérification des faits	23 921	2

Cette section a présenté un aperçu détaillé des différents ensembles de données utilisés dans cette étude pour la détection des fausses nouvelles. En mettant en lumière leurs caractéristiques spécifiques, tels que le domaine d'application, les méthodes de vérification des faits et la taille des corpus, nous avons démontré l'importance d'utiliser des jeux de données variés pour évaluer la robustesse des modèles de détection. La diversité des domaines couverts, ainsi que les différences en termes de longueur des articles et de méthodes de vérification des faits, permettent d'illustrer les défis inhérents à la détection des fausses nouvelles. L'analyse comparative des performances sur ces ensembles de données fournira des indications précieuses sur l'efficacité des approches basées sur le texte dans des contextes variés.

3.2 Optimiseurs utilisés en détection de fausses nouvelles

Les optimiseurs sont des composants essentiels dans les modèles d'apprentissage automatique, car ils guident l'ajustement des paramètres du modèle, généralement les poids et biais, pour minimiser une fonction de perte donnée. Dans le cadre de la détection des fausses nouvelles, qui repose fortement sur des modèles d'apprentissage automatique et des techniques de traitement du langage naturel (NLP), le choix d'un optimiseur efficace est essentiel pour améliorer les performances du modèle, la vitesse d'entraînement et la capacité de généralisation. Plusieurs optimiseurs sont largement utilisés dans les modèles de détection des fausses nouvelles à la pointe de la technologie, en particulier ceux basés sur l'apprentissage profond et les réseaux de neurones. Cette section offre un aperçu détaillé des optimiseurs les plus importants utilisés dans les tâches de détection des fausses nouvelles, en discutant de leurs mécanismes, avantages et limitations.

3.2.1 Descente de Gradient Stochastique

Le gradient stochastique de descente est l'un des algorithmes d'optimisation les plus essentiels employés dans le domaine de l'apprentissage automatique. En calculant le gradient de la fonction de perte par rapport à un petit sous-ensemble (ou lot) de données d'entraînement, elle met à jour les paramètres du modèle.

Mécanisme : La règle de mise à jour des poids θ dans SGD est donnée par :

$$\theta = \theta - \eta \nabla_\theta J(\theta) \tag{3.1}$$

où η est le taux d'apprentissage et $J(\theta)$ est la fonction de perte.

Avantages :

— **Efficacité** : En utilisant un petit lot de données plutôt que l'ensemble du jeu de données, SGD accélère le processus d'entraînement, ce qui le rend très adapté aux jeux de données à grande échelle, tels que ceux utilisés dans la détection des fausses nouvelles.

— **Généralisation** : Le bruit inhérent aux mises à jour (puisque le gradient est calculé sur un sous-ensemble aléatoire) aide à empêcher le modèle de surajuster les données d'entraînement.

Inconvénients :

— **Convergence lente** : SGD peut être lent à converger, surtout pour les modèles avec des données de grande dimension, ce qui est courant dans les tâches de détection de fausses nouvelles basées sur du texte.

— **Sensibilité au taux d'apprentissage** : Le taux d'apprentissage doit être soigneusement ajusté. Un taux d'apprentissage trop élevé peut entraîner des mises à jour instables, tandis qu'un taux d'apprentissage trop faible peut ralentir l'entraînement.

Applications en détection de fausses nouvelles : SGD est couramment utilisé dans les réseaux de neurones simples et les modèles de régression logistique qui effectuent des tâches de classification de texte. Dans la détection des fausses nouvelles, il sert de technique d'optimisation de base, en particulier lorsque les ressources de calcul sont limitées ou lorsque des modèles plus simples sont utilisés comme références.

3.2.2 Adam (Estimation des Moments Adaptatifs)

Adam est l'un des optimiseurs les plus populaires dans le domaine de l'apprentissage profond, y compris la détection des fausses nouvelles. Il combine les avantages de deux autres optimiseurs : AdaGrad et RMSProp, en adaptant le taux d'apprentissage pour chaque paramètre en fonction des moments des gradients.

Mécanisme : Adam calcule une moyenne exponentiellement décroissante des gradients passés (premier moment) et du carré des gradients passés (second moment). La règle de mise à jour des paramètres θ est :

$$\theta = \theta - \eta \frac{m_t}{\sqrt{v_t} + \varepsilon} \tag{3.2}$$

où m_t est le premier moment (moyenne des gradients), v_t est le second moment (variance des gradients), et ε est une petite constante pour éviter la division par zéro.

Avantages :

— **Taux d'apprentissage adaptatif** : Adam ajuste automatiquement le taux d'apprentissage pour chaque paramètre, ce qui le rend très efficace pour les modèles complexes et les grands jeux de données, comme ceux utilisés pour la détection des fausses nouvelles.

— **Convergence rapide** : Adam converge plus rapidement que SGD classique, en particulier dans les modèles basés sur du texte où les jeux de données sont volumineux et de grande dimension.

— **Robustesse** : Adam fonctionne bien avec une large gamme d'architectures de modèles et de jeux de données sans nécessiter un ajustement important des hyperparamètres.

Inconvénients :

— **Utilisation de la mémoire** : Adam nécessite de stocker à la fois les estimations du premier et du second moment pour chaque paramètre, ce qui peut être gourmand en mémoire, en particulier dans les modèles de grande envergure comme les transformateurs.

— **Surapprentissage** : Bien qu'Adam converge rapidement, il peut parfois entraîner un surapprentissage, surtout lorsque la capacité du modèle est élevée, comme dans les réseaux de neurones profonds ou les modèles basés sur des transformateurs pour la détection des fausses nouvelles.

Applications en détection de fausses nouvelles : Adam est l'optimiseur de choix pour les modèles d'apprentissage profond dans la détection des fausses nouvelles. Il est largement utilisé dans les réseaux de CNN,RNN,LSTM ainsi que les modèles basés sur les transformateurs comme BERT . Ces modèles reposent sur Adam pour ajuster efficacement les paramètres, ce qui leur permet d'obtenir une précision élevée dans les tâches de classification de texte, y compris la détection des fausses nouvelles.

3.2.3 RMSProp (Root Mean Square Propagation)

RMSProp est un autre optimiseur conçu pour résoudre les problèmes de la descente de gradient stochastique, en particulier en ce qui concerne l'ajustement du taux d'apprentissage. Il s'agit d'une méthode de taux d'apprentissage adaptatif qui divise le taux d'apprentissage par une moyenne mobile des gradients carrés.

Mécanisme : La règle de mise à jour de RMSProp est :

$$\theta = \theta - \frac{\eta}{\sqrt{E[g^2]_t + \varepsilon}} \nabla_\theta J(\theta) \tag{3.3}$$

où $E[g^2]_t$ est la moyenne pondérée exponentiellement des gradients carrés passés.

Avantages :

— **Convergence stable** : En ajustant le taux d'apprentissage en fonction de la magnitude récente des gradients, RMSProp évite les oscillations importantes qui peuvent survenir dans SGD, assurant une convergence plus stable.

— **Adapté aux problèmes non stationnaires** : RMSProp est particulièrement efficace dans les cas où le paysage de la perte change avec le temps, ce qui est souvent le cas dans les tâches de NLP lorsque le modèle apprend à différencier les articles de fausses nouvelles des véritables.

Inconvénients :

— **Nécessité de réglage** : Bien que RMSProp ajuste le taux d'apprentissage, il nécessite tout de même un certain ajustement, notamment en ce qui concerne son facteur de décroissance et son taux d'apprentissage.

Applications en détection de fausses nouvelles : RMSProp est utilisé dans les modèles où les mises à jour des gradients doivent être stabilisées, comme dans les RNNs et les LSTMs. Ces modèles traitent les séquences de texte pour la détection des fausses nouvelles et nécessitent un équilibre entre un apprentissage efficace et la prévention des gradients explosifs ou disparus. RMSProp aide à garantir un entraînement stable dans de tels contextes.

3.2.4 AdaGrad (Adaptive Gradient Algorithm)

AdaGrad ajuste le taux d'apprentissage pour chaque paramètre en fonction de l'historique des gradients. Les paramètres qui reçoivent des mises à jour fréquentes verront leur taux d'apprentissage réduit, tandis que les paramètres mis à jour de manière moins fréquente auront un taux d'apprentissage plus élevé.

Mécanisme : La règle de mise à jour d'AdaGrad est :

$$\theta = \theta - \frac{\eta}{\sqrt{G_t + \varepsilon}} \nabla_\theta J(\theta) \tag{3.4}$$

où G_t est la somme des carrés des gradients pour le paramètre θ jusqu'à l'instant t.

Avantages :

— **Apprentissage adaptatif** : AdaGrad permet à chaque paramètre d'avoir son propre taux d'apprentissage, ce qui le rend adapté aux données rares,comme cela se produit fréquemment dans les jeux de données textuels.

— **Bon pour les gradients rares** : Dans la détection des fausses nouvelles, certains mots ou modèles peuvent être rares mais cruciaux pour la détection. La capacité d'AdaGrad à accorder plus d'attention aux caractéristiques moins fréquentes le rend adapté à ces tâches.

Inconvénients :

— **Décroissance agressive du taux d'apprentissage** : L'une des limites majeur d'AdaGrad réside dans sa tendance à réduire rapidement et fortement le taux d'apprentissage au cours du temps. Cette diminution trop rapide risque d'interrompre le processus d'apprentissage du modèle avant qu'il n'ait atteint son plein potentiel.

Applications en détection de fausses nouvelles : AdaGrad est souvent utilisé dans les modèles basés sur le texte traitant des représentations de mots ou des caractéristiques rares. Il peut ne pas être aussi populaire qu'Adam pour les modèles à grande échelle, mais il est précieux dans les situations où la rareté des données est significative.

3.2.5 Analyse comparative

Dans les tâches de détection des fausses nouvelles, le choix de l'optimiseur peut avoir un impact significatif à la fois sur les performances du modèle et sur l'efficacité de l'entraînement. Adam se distingue comme l'optimiseur le plus largement utilisé, en particulier pour les modèles basés sur l'apprentissage profond, tels que les CNN, RNN et les modèles basés sur des transformateurs comme BERT. Son taux d'apprentissage adaptatif et sa convergence rapide le rendent adapté au traitement de la complexité et de l'ampleur des jeux de données liés à la détection des fausses nouvelles. Cependant, des modèles plus simples ou des cas où l'efficacité mémoire est une préoccupation peuvent encore bénéficier de l'utilisation de SGD ou RMSProp. Les modèles traitant des caractéristiques rares, souvent rencontrés dans les jeux de données textuels, peuvent voir leurs performances améliorées avec AdaGrad.

Chaque optimiseur a ses points forts et ses compromis, et le choix final dépend souvent des caractéristiques spécifiques de la tâche de détection des fausses nouvelles. Par exemple, la robustesse et l'efficacité d'Adam en font un choix idéal pour traiter des données volumineuses et de grande dimension, tandis que RMSProp offre plus de stabilité dans les modèles basés sur des séquences comme les LSTM, qui sont courants dans les tâches de NLP. Dans les futures recherches, des techniques d'optimisation hybrides ou de nouveaux optimiseurs pourraient émerger, améliorant encore la capacité des modèles à détecter les fausses nouvelles avec une plus grande précision et efficacité.

3.3 Métriques d'évaluation

Dans cette section, nous abordons les principales métriques d'évaluation utilisées pour juger les performances des modèles de détection des fausses nouvelles. La détection de fausses nouvelles repose généralement sur des modèles de classification qui visent à distinguer les articles de presse réels des articles trompeurs. Pour évaluer la qualité des prédictions effectuées par ces modèles, différentes mesures standard sont utilisées, comme l'exactitude, la précision, le rappel et le score F1. En outre, des outils visuels comme la Matrice de Confusion et la courbe ROC sont utilisés pour mieux comprendre

la performance des modèles et leur capacité à gérer les jeux de données déséquilibrés. Nous allons explorer en détail ces métriques afin de mettre en lumière leur rôle dans l'analyse des modèles de détection.

La plupart des approches de détection des fausses nouvelles construisent un modèle de classification pour prédire si un article de presse appartient à la classe réelle ou à la classe fausse.

Matrice de Confusion est une matrice carrée qui regroupe toutes les classes en directions horizontale et verticale. . La liste des classes en haut représente les sorties prédites, et la liste sur le côté gauche représente les cibles. Les éléments de la matrice en position (i, j) indiquent combien de points de données avec l'étiquette de classe C_i sont mal classés en classe C_j.

Exactitude est définie par la formule suivante :

$$\text{Exactitude} = \frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} \tag{3.5}$$

où TP, TN, FP et FN désignent respectivement les Vrais Positifs, Vrais Négatifs, Faux Positifs et Faux Négatifs. L'exactitude mesure la proportion des prédictions justes par rapport à l'ensemble des échantillons analysés.

Précision La Précision est définie par la formule suivante :

$$\text{Précision} = \frac{\#TP}{\#TP + \#FP} \tag{3.6}$$

Elle représente le ratio de toutes les fausses nouvelles identifiées par rapport à celles annotées comme fausses nouvelles.

Rappel Le Rappel est défini par la formule suivante :

$$\text{Rappel} = \frac{\#TP}{\#TP + \#FN} \tag{3.7}$$

Il représente le ratio des fausses nouvelles par rapport aux résultats prédits.

Score F1 Le Score F1 est défini par la formule suivante :

$$\text{Score F1} = \frac{2 \cdot \text{Précision} \cdot \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{3.8}$$

Il donne la performance globale de la prédiction.

Spécificité La Spécificité, également appelée Taux de Vrais Négatifs, est définie par la formule suivante :

$$\text{Spécificité} = \frac{\#TN}{\#TN + \#FP} \tag{3.9}$$

Elle est liée à la capacité du classificateur à identifier les résultats négatifs.

Courbe ROC La courbe ROC compare les performances des classificateurs en observant le compromis entre le Taux de Faux Positifs et le Taux de Vrais Positifs , où

$$\text{TPR} = \frac{\#TP}{\#TP + \#FN} \quad \text{et} \quad \text{FPR} = \frac{\#FP}{\#FP + \#TN} \tag{3.10}$$

À partir de la courbe ROC, la Surface Sous la Courbe (AUC) montre la performance globale du classificateur.

$$\text{AUC} = \frac{(n_0 + n_1 + 1 - r_i) - n_0(n_0+1)/2}{n_0 n_1} \tag{3.11}$$

où r_i est le rang du i^{me} article de fausse nouvelle et n_0 (n_1) est le nombre de fausses (vraies) nouvelles. Le score AUC est plus cohérent et discriminant que l'exactitude et est principalement utilisé pour traiter les problèmes de classification déséquilibrés. Il est intéressant de noter que la plupart des ensembles de données utilisés pour la détection des fausses nouvelles sont connus pour leur distribution déséquilibrée. Le tableau 3.3 résume les expériences récentes sur la détection des fausses nouvelles.

TABLE 3.3 – Expériences récentes sur la détection des fausses nouvelles

Étude	**Modèle d'apprentissage profond**	**Ensemble de données**	**Exactitude précision rappel score F1**
Agarwal et al. [84]	CNN+LSTM	Kaggle fausses nouvelles	Précision 97,26%
Huang et al. [199]	Méthode d'ensemble	Satire	Exactitude 99,4%
Kaliyar et al. [85]	CNN profond	Kaggle fausses nouvelles	Exactitude 98,36%
Fang et al. [200]	CNN	MIT fausses nouvelles	Rappel 95,6%
Dong et al. [201]	CNN à deux chemins	PHEME	Rappel macro 77,58%
Khattar et al. [202]	Autoencodeur variationnel	Weibo & MediaEval	Exactitude 82,4%
Li et al. [185]	CNN	Twitter15 & Weibo	Exactitude moyenne 91,67%
Li et al. [185]	CNN	LIAR & KaggleFN	Exactitude moyenne 92,08%
Guo et al. [203]	GRU	Weibo	Exactitude 87,2%
Wu et al. [204]	Auto-attention contrôlée	RumorEval	F1 82,19%
Zhou et al. [205]	CNN multimodal	PolitiFact	Exactitude 87,4%
Schwarz et al. [206]	Transformateur encodeur	Twitter	Exactitude 94,08%

L'évaluation des modèles de détection des fausses nouvelles repose sur un ensemble varié de métriques qui permettent de juger de leur efficacité et de leur précision. Des indicateurs comme l'Exactitude, la Précision, le Rappel et le Score F1 fournissent une évaluation globale des performances des modèles dans différents contextes. De plus, la courbe ROC et le score AUC permettent de mieux comprendre les compromis entre les Faux Positifs et les Vrais Positifs, particulièrement utiles dans les scénarios de classification déséquilibrée. Enfin, les résultats des expériences récentes sur différents modèles, comme illustré dans le tableau 3.3, montrent l'évolution continue des techniques d'apprentissage profond dans

la détection des fausses nouvelles, offrant des perspectives intéressantes pour les futures recherches dans ce domaine.

3.4 Études Comparatives entre CNN, LSTM,BI-LSTM,HAN, les HAN convolutifs, ainsi que le classificateur Naive Bayes

La plupart des travaux antérieurs sur la détection des fausses nouvelles ont appliqué plusieurs méthodes traditionnelles d'apprentissage automatique et de réseaux de neurones pour détecter les fausses informations. Cependant, ces travaux se sont concentrés sur la détection d'informations de types particuliers, comme les politiques, et ont développé leurs modèles pour des ensembles de données spécifiques. Il est probable que ces approches souffrent de biais dans les ensembles de données et qu'elles performent mal sur des nouvelles d'un autre sujet. De plus, une limitation majeure des études comparatives précédentes est qu'elles sont menées sur un type spécifique d'ensemble de données, ce qui rend difficile de tirer une conclusion sur la performance des divers modèles. En outre, ces articles se sont souvent concentrés sur un nombre limité de caractéristiques, entraînant une exploration incomplète des caractéristiques potentielles des fausses nouvelles.

Nous avons constaté que la plupart des travaux connexes se concentrent sur l'amélioration de la qualité de la prédiction en ajoutant des fonctionnalités supplémentaires. Le problème est que ces fonctionnalités ne sont pas toujours disponibles ; par exemple, certains articles peuvent ne pas contenir d'images. De plus, l'utilisation d'informations provenant des réseaux sociaux pose problème car il est facile de créer un nouveau compte et de tromper le système de détection. D'où notre décision de se focaliser exclusivement sur le corps de l'article et de voir s'il est possible de détecter avec précision les fausses nouvelles en utilisant uniquement le contenu textuel. Comme cela a été montré dans les sections précédentes, plusieurs approches peuvent être utilisées pour extraire des caractéristiques textuelles et les utiliser dans des modèles. Cela met l'accent sur la fonctionnalité du contenu textuel des nouvelles.

3.4.1 Vue d'ensemble de l'approche

Notre approche initiale pour la détection des fausses nouvelles, comme illustré à la Figure 3.2, commence par une phase essentielle de prétraitement des données. Cette étape implique la suppression des caractères et mots inutiles, ainsi que la normalisation des données textuelles pour garantir une meilleure qualité et cohérence des informations traitées. Une fois les données nettoyées, nous procédons à l'extraction des entités N-gram, qui sont des séquences continues de n éléments (souvent des mots) extraites du texte. Ces entités N-gram permettent de capturer les motifs de langage qui peuvent être significatifs pour la classification des documents.

Nous formons ensuite une matrice d'entités, représentant les documents par des vecteurs de caractéristiques dérivées des N-grams. Cette matrice constitue la base sur laquelle les modèles de classification vont opérer. La phase suivante du processus est l'entraînement du classificateur. Nous avons évalué

plusieurs algorithmes de classification pour prédire la classe des documents, incluant six algorithmes d'apprentissage profond :CNN, LSTM,BI-LSTM,HAN, les HAN convolutifs, ainsi que le classificateur Naive Bayes.

Pour la mise en œuvre, nous avons utilisé des bibliothèques Python, en particulier le Natural Language Toolkit, pour intégrer ces classificateurs dans notre pipeline. Afin d'évaluer correctement la performance de chaque modèle, nous avons divisé notre jeu de données en ensembles d'entraînement et de test. En pratique, environ 80% des données ont été utilisées pour l'entraînement des modèles, tandis que les 20% restants ont été réservés pour les tests lors de chaque cycle de validation. Cette approche permet d'assurer que les modèles sont évalués de manière rigoureuse et que leurs performances sont mesurées de façon fiable sur des données non vues pendant l'entraînement.

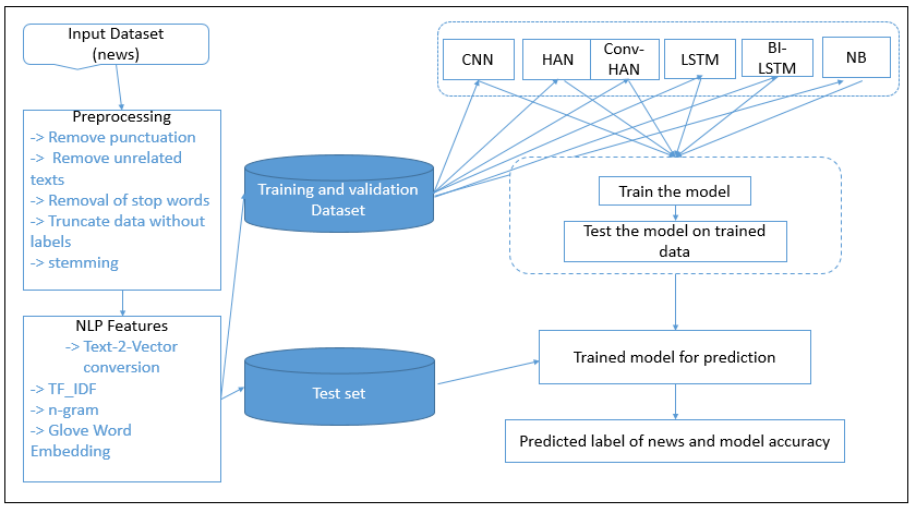


FIGURE 3.2 – Organigramme du processus proposé de détection des fausses nouvelles [7]

3.4.2 Évaluation Expérimentale

Informations Statistiques sur les Ensembles de Données

Les modèles ont été testés sur quatre ensembles de données différents : (i) Data an Open Sources, un ensemble de données contenant 9 408 908 articles dont 11 161 articles de catégories fausses et fiables ont été sélectionnés. (ii) L'ensemble de données Fake or Real News est développé par George McIntire. La partie fausse de cet ensemble de données a été collectée à partir de l'ensemble de données de fausses nouvelles de Kaggle, incluant les nouvelles du cycle électoral de 2016 aux États-Unis. (iii) Liar est un ensemble de données accessible au public [31]. Il comprend 12,8K de courtes déclarations humaines provenant de l'API POLITIFACT.COM. Il comprend six étiquettes de véracité : pants-fire, false, hardly true, half true, almost true, et true. Dans notre travail, nous essayons de différencier les vraies nouvelles de tous les types de canulars, propagandes, satires et informations trompeuses. Par conséquent, nous nous concentrons principalement sur la classification des informations comme réelles et fausses. (iv) L'ensemble de données True / Fake est un ensemble de données accessible au public. La partie fausse

contient 17 903 nouvelles réparties entre 5 étiquettes : news politics, government news, leftnews usnews et middle east. La partie vraie contient 20 826 nouvelles de deux étiquettes : news politics et worldnews. Ces données sont extraites des nouvelles entre le 13 janvier 2016 et le 31 décembre 2018.

Caractéristiques

Pour construire un modèle d'apprentissage profond, la sélection des caractéristiques est d'une importance capitale pour des performances optimales du système. Les caractéristiques utilisées dans le modèle proposé sont les suivantes :

Intégration de mots pré-entraînés L'intégration de mots est une méthode pour représenter les mots et les documents en utilisant des vecteurs denses. Contrairement aux approches traditionnelles de codage telles que le sac de mots, qui utilisaient de grands vecteurs épars pour représenter chaque mot ou chaque mot dans un vecteur pour couvrir l'ensemble du vocabulaire, les représentations vectorielles denses sont plus compactes. Dans ces schémas traditionnels, les vecteurs étaient principalement constitués de zéros en raison de la taille importante du vocabulaire, ce qui rendait la représentation moins efficace.

Avec l'intégration de mots, chaque mot est représenté par un vecteur dense, ce qui correspond à une projection du mot dans un espace vectoriel continu. La position d'un mot dans cet espace est déterminée par le contexte dans lequel il apparaît, c'est-à-dire les mots qui l'entourent. Cette position, obtenue par apprentissage à partir du texte, est ce que l'on appelle l'intégration du mot. Deux exemples de méthodes pour apprendre à intégrer des mots à partir de textes : Word2Vec et GloVe. En plus de ces méthodes soigneusement conçues, l'intégration de mots peut être apprise dans le cadre d'un modèle d'apprentissage profond. Cela peut être une approche plus lente, mais elle permet d'adapter le modèle à un ensemble de données d'entraînement spécifique.

— GloVe : GloVe est un algorithme d'apprentissage non supervisé qui permet de découvrir la proximité de deux mots, avec leur séparation dans l'espace vectoriel. Ces représentations vectorielles créées sont appelées vecteurs d'intégration de mots. Dans GloVe, l'entraînement est effectué sur les statistiques de co-occurrence globale des mots à partir d'un corpus. Dans le cas de GloVe, la matrice de comptage est prétraitée en normalisant les comptes et en les lissant via un logarithme. GloVe permet une implémentation parallèle, ce qui facilite l'entraînement sur davantage de données.

— Word2Vec : Word2Vec est disponible en deux modes : sac de mots continu (CBOW) et skip-gram. Il a été initialement conçu pour prédire un mot dans un contexte. Par exemple, étant donné deux mots précédents et les deux mots suivants, quel mot est le plus susceptible de se produire entre eux. Mais il semble que la représentation cachée de ces mots fonctionne bien comme intégration de mots et possède des propriétés très intéressantes telles que les mots avec un sens similaire ont une représentation vectorielle similaire. Il est également possible d'effectuer des calculs arithmétiques qui capturent des informations telles que le singulier, le pluriel ou même les capitales et les pays.

Caractéristiques de Comptage des n-grammes Ces caractéristiques sont utilisées pour compter les occurrences de n-grammes dans le titre et le corps des nouvelles, et divers ratios du n-gram unique et du nombre total de mots donné par l'Éq. (1).

$$ratio\,de\,ngram\,unique = \frac{total\,ngram\,unique}{total\,ngram} \tag{3.12}$$

Sac de Mots La technique du sac de mots (BoW) traite chaque article de presse comme un document et calcule le nombre de fréquences de chaque mot dans ce document, qui est ensuite utilisé pour créer une représentation numérique des données, également connue sous le nom de caractéristiques vectorielles de longueur fixe. Le sac de mots convertit le texte brut en vecteur de comptage des mots avec la fonction CountVectorizer pour l'extraction des caractéristiques. CountVectorizer divise le contenu du texte, construit le vocabulaire et encode le texte en un vecteur. Ce vecteur encodé aura un comptage des occurrences de chaque mot qui ressemble davantage à un comptage de fréquence sous forme de paire clé/valeur. Cette méthodologie présente des inconvénients en termes de perte d'information. La position relative des mots est ignorée et l'information contextuelle est perdue.

Tf-Idf est une méthode de représentation des caractéristiques statistiques en traitement du langage naturel pour évaluer et pondérer les mots afin d'identifier les mots pertinents et importants dans un document. Cette méthode a été utilisée dans de nombreux problèmes de classification de texte, notamment pour la détection des fausses nouvelles- La formule pour déterminer la Fréquence Terme est la suivante :

$$TF = \frac{nombre\,de\,fois\,que\,le\,terme\,apparat\,dans\,un\,document}{nombre\,total\,de\,mots\,dans\,le\,document} \tag{3.13}$$

$$IDF = log(\frac{nombre\,de\,documents\,dans\,le\,corpus}{nombre\,de\,documents\,dans\,le\,corpus\,contenant\,le\,terme}) \tag{3.14}$$

Le TF-IDF d'un terme est calculé en multipliant les scores TF et IDF.

$$TF - IDF = \text{TF} \times \text{IDF} \tag{3.15}$$

3.4.3 Extraction des Caractéristiques et Implémentation du Modèle

Prétraitement

Les données textuelles nécessitent un prétraitement spécial pour les implémenter dans des algorithmes d'apprentissage automatique ou d'apprentissage profond. Il existe diverses techniques largement utilisées pour convertir les données textuelles en une forme prête pour la modélisation. Les étapes de prétraitement des données que nous décrivons ci-dessous sont appliquées au contenu des nouvelles. Les différentes représentations de vecteurs de mots que nous avons employées dans le cadre de notre analyse sont également données.

— **Nuage de mots :** Avant de commencer le prétraitement, nous visualisons nos données à partir du nuage de mots des mots-clés les plus utilisés dans nos données.

— **Suppression de la Ponctuation :** La ponctuation dans le langage naturel fournit le contexte grammatical de la phrase. Les signes de ponctuation tels qu'une virgule peuvent ne pas ajouter beaucoup de valeur à la compréhension du sens de la phrase.

— **Suppression des mots vides :** Nous commençons par supprimer les mots vides des données textuelles disponibles. Les mots vides sont des mots insignifiants dans une langue qui créeront du bruit lorsqu'ils seront utilisés comme caractéristiques dans la classification de texte. Les mots les plus courants dans une langue qui ne fournissent pas beaucoup de contexte peuvent être traités et filtrés du texte car ils sont plus courants et contiennent moins d'informations utiles. Nous avons utilisé la bibliothèque Natural Language Toolkit (NLTK) pour supprimer les mots vides.

— **Racination :** La racination est une technique de suppression des préfixes et des suffixes d'un mot, se terminant par la racine. En utilisant la racine, nous pouvons réduire les formes fléchies et parfois les formes dérivées d'un mot à une forme de base commune.

Extraction des Caractéristiques

La performance des modèles d'apprentissage profond dépend en grande partie de la conception des caractéristiques.

— **Extraction des caractéristiques n-grammes :** Le n-gram basé sur les mots a été utilisé pour représenter le contexte du document et générer des fonctionnalités pour classer le document comme faux et réel. De nombreux travaux existants ont utilisé les approches unigramme (n = 1) et bigramme (n = 2) pour la détection des fausses nouvelles [207]. Nous avons utilisé la fonction TfidfVectorizer de la bibliothèque sklearn d'extraction de caractéristiques en Python pour générer des fonctionnalités TF-IDF n-grammes.

— **Intégration de Mots Pré-entraînée :** Pour les modèles de réseaux de neurones, les intégrations de mots ont été initialisées avec des intégrations pré-entraînées de GloVe en 100 dimensions [114]. Leur entraînement s'est déroulé sur un ensemble de données comprenant un milliard de tokens (mots) et un vocabulaire de 400 000 mots. Dans cette architecture LSTM avec gensim, nous avons utilisé Word2Vec de Google pour représenter les mots dans des intégrations d'espace vectoriel de 100 dimensions.

— **Sac de Mots (BoW) :** La technique du sac de mots traite chaque article de presse comme un document et calcule le nombre de fréquences de chaque mot dans ce document, qui est ensuite utilisé pour créer une représentation numérique des données, également connue sous le nom de caractéristiques vectorielles de longueur fixe. Le sac de mots convertit le texte brut en vecteur de comptage des mots avec la fonction CountVectorizer pour l'extraction des caractéristiques. CountVectorizer divise le contenu du texte, construit le vocabulaire, et encode le texte en un vecteur. Ce vecteur encodé aura un comptage des occurrences de chaque mot qui ressemble davantage à un

comptage de fréquence sous forme de paire clé/valeur. Cette méthodologie présente des inconvénients en termes de perte d'information. La position relative des mots est ignorée et l'information contextuelle est perdue.

Implémentation des Approches

Dans cette section, nous décrivons la configuration expérimentale des différents modèles basés sur les réseaux de neurones et l'apprentissage profond utilisés dans notre expérience. Nous fournissons également des détails sur l'implémentation de nos nouvelles approches explorées dans la détection des fausses nouvelles.

CNN Le modèle de Réseaux de Neurones Convolutifs a été initialisé comme une séquence de couches. Nous utiliserons une structure de réseau entièrement connecté avec trois couches. Les couches entièrement connectées sont définies en utilisant la classe Dense. Le premier argument peut être le nombre de neurones ou de nœuds présents dans la couche, tandis que l'argument activation permet de spécifier la fonction d'activation. Les deux premières couches seront utilisées avec la fonction d'activation ReLU (rectified linear unit) et la couche de sortie sera utilisée avec la fonction sigmoïde. Nous employons une sigmoïde sur la couche de sortie afin de garantir que notre sortie de réseau est comprise entre 0 et 1 et facilement compatible avec une probabilité de classe 1 ou une classification complexe de chaque classe, avec un seuil par défaut de 0,5. L'optimiseur ADAM a été utilisé pour compiler le modèle, avec un taux d'apprentissage de 0,001, afin de réduire au minimum la perte d'entropie croisée binaire. Finalement, ce modèle a été développé à travers trois époques.

LSTM **LSTM + gensim** Nous utilisons la bibliothèque gensim en Python qui prend en charge une multitude de classes pour les applications de traitement du langage naturel. Comme discuté, nous utilisons un modèle CBOW avec un échantillonnage négatif et des vecteurs de mots de 100 dimensions. L'optimiseur ADAM avec un taux d'apprentissage de 0,001 a été appliqué pour minimiser la perte d'entropie croisée binaire et la sigmoïde était la fonction d'activation pour la couche de sortie finale. Enfin, ce modèle a été entraîné sur 3 époques avec des lots de 64 et 512.

LSTM + glove Le modèle LSTM a été pré-entraîné avec des intégrations GloVe en 100 dimensions. La dimension de sortie et les pas de temps ont été réglés à 300. L'optimiseur ADAM avec un taux d'apprentissage de 0,001 a été appliqué pour minimiser la perte d'entropie croisée binaire et la sigmoïde était la fonction d'activation pour la couche de sortie finale. Enfin, ce modèle a été entraîné sur 3 époques avec des lots de 64 et 512.

Bi-LSTM Le but du modèle Bi-LSTM est de détecter les anomalies dans une certaine partie des nouvelles, nous devons l'examiner avec les événements d'action précédents et suivants. Bi-LSTM a été initialisé avec des intégrations GloVe pré-entraînées de 100 dimensions. Une dimension de sortie de 100 et des pas de temps de 300 ont été appliqués. L'optimiseur ADAM avec un taux d'apprentissage de 0,001

a été utilisé pour minimiser la perte d'entropie croisée binaire. La taille du lot d'apprentissage a été fixée à 128 et une perte à chaque époque a été observée avec le rappel.

HAN Le réseau d'attention hiérarchique consistait en deux mécanismes d'attention pour le codage au niveau des mots et des phrases. Avant l'entraînement, nous avons fixé le nombre maximum de phrases dans un article de presse à 20 et le nombre maximum de mots dans une phrase à 100. Dans les deux niveaux de codage, un GRU bidirectionnel avec une dimension de sortie de 100 a été introduit dans notre couche d'attention personnalisée. Nous avons utilisé un encodeur de mots comme entrée de notre couche temporelle d'encodeur de phrases distribuée. Nous avons optimisé notre modèle avec ADAM qui a appris à un taux de 0,001. Nous avons utilisé la bibliothèque Keras avec le backend Tensorflow pour implémenter le mécanisme d'attention.

HAN Convolutionnel Afin d'extraire des caractéristiques d'entrée de haut niveau, nous avons incorporé une couche convolutive unidimensionnelle avant chaque couche GRU bidirectionnelle dans HAN. Cette couche a sélectionné les caractéristiques de chaque trigramme de l'article de presse avant de les passer à la couche d'attention.

Naive Bayes Nous avons également exploré des modèles traditionnels d'apprentissage automatique utilisant des techniques de traitement du langage naturel. Nous supprimons les suffixes des mots en les dérivant avec le Snowball Stemmer de la bibliothèque NLTK. Le CountVectorizer de Scikit-Learn fournit un moyen simple de tokeniser une collection de documents texte et de créer un vocabulaire de mots connus, ainsi que d'encoder de nouveaux documents en utilisant ce vocabulaire. Ainsi, dans le CountVectorizer, nous utilisons le comptage des mots, dans TFIDF. Nous avons utilisé le classificateur multinomial, Naive Bayes, comme modèle suivant. Nous y avons introduit les caractéristiques n-grammes. Nous avons utilisé la fonction de bibliothèque Python nommée MultinomialNB pour cela.

3.4.4 Résultat

Dans cette section, nous décrivons une analyse des performances de nos modèles basés sur les réseaux de neurones et l'apprentissage profond. Nous présentons les meilleures performances pour chaque ensemble de données. Nous calculons l'exactitude, la précision, le rappel et le score F1 pour les classes fausses et réelles, et trouvons leur moyenne, média pondéré (le nombre d'instances vraies pour chaque classe) et rapportons un score moyen de ces métriques.

Métriques d'Évaluation

Nous utilisons l'exactitude, la précision, le rappel et le score F1 comme métriques d'évaluation (tp, fp, fn dans les

équations suivantes sont respectivement les vrais positifs, faux positifs et faux négatifs). La précision est une mesure calculée comme le ratio des prédictions correctes au nombre total d'exemples. La

précision mesure le pourcentage de prédictions positives qui sont correctes et est définie comme suit :

$$Precision = \frac{tp}{tp + fp} \tag{3.16}$$

Le rappel consiste à mesurer le pourcentage de prédictions correctes que le classificateur capture et est défini comme suit :

$$Rappel = \frac{tp}{tp + fn} \tag{3.17}$$

Le score F1 permet de trouver l'équilibre entre le rappel et la précision et est calculé comme suit :

$$ScoreF1 = \frac{\text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{3.18}$$

TABLE 3.4 – Résultats des modèles sur différents ensembles de données

Modèles		Vrai / Faux				Entraînement / Test				Données d'actualités				Fausses ou vraies nouvelles			
		Préc	Précision	Rappel	Score F1	Préc	Précision	Rappel	Score F1	Préc	Précision	Rappel	Score F1	Préc	Précision	Rappel	Score F1
LSTM	word2vec (Gensim)	.98	.98	.98	.98	.56	.78	.50	.36	.96	.96	.96	.96	.86	.86	.86	.86
LSTM	Glove Embedding	.98	.98	.98	.98	.54	.27	.50	.35	.75	.75	.75	.75	.74	.75	.74	.74
BiLSTM	Glove Embedding	.78	.50	.78	.89	.61	.61	.59	.59	.84	.84	.84	.84	.84	.84	.84	.84
HAN	Glove Embedding	.59	.59	.59	.59	.57	.57	.57	.57	.94	.94	.94	.94	.86	.86	.86	.86
Conv-HAN	Glove Embedding	.56	.56	.56	.56	.59	.59	.59	.59	.83	.83	.83	.83	.80	.82	.80	.80
CNN	TF-idf Countvectorizer	.90	.90	.90	.90	.97	.97	.97	.97	.77	.83	.78	.76	.79	.84	.80	.79
Naive Bayes	TF-idf Countvectorizer	.97	.97	.97	.97	.83	.87	.84	.83	.97	.97	.97	.97	.90	.90	.90	.90

Résultats et discussion

Comme indiqué dans le Tableau 1, le modèle basé sur le mécanisme d'attention est le plus vulnérable au surapprentissage sur les ensembles de données True / Fake et Liar. Bien que BI-LSTM soit également victime de surapprentissage sur tous les ensembles de données, il montre ses meilleures performances sur l'ensemble de données True / Fake. Les modèles utilisés avec succès pour la classification de texte comme LSTM, Bi-LSTM, HAN, Conv-HAN peinent à surmonter le problème de surapprentissage pour l'ensemble de données Liar. Les modèles CNN et Naive Bayes montrent les meilleures performances parmi les modèles avec une précision de 80%. Le modèle NB atteint plus de 90% de précision et un score F1 supérieur à 0.97. Ce résultat indique que bien que les modèles basés sur les réseaux de neurones puissent souffrir de surapprentissage pour un petit ensemble de données (LIAR).

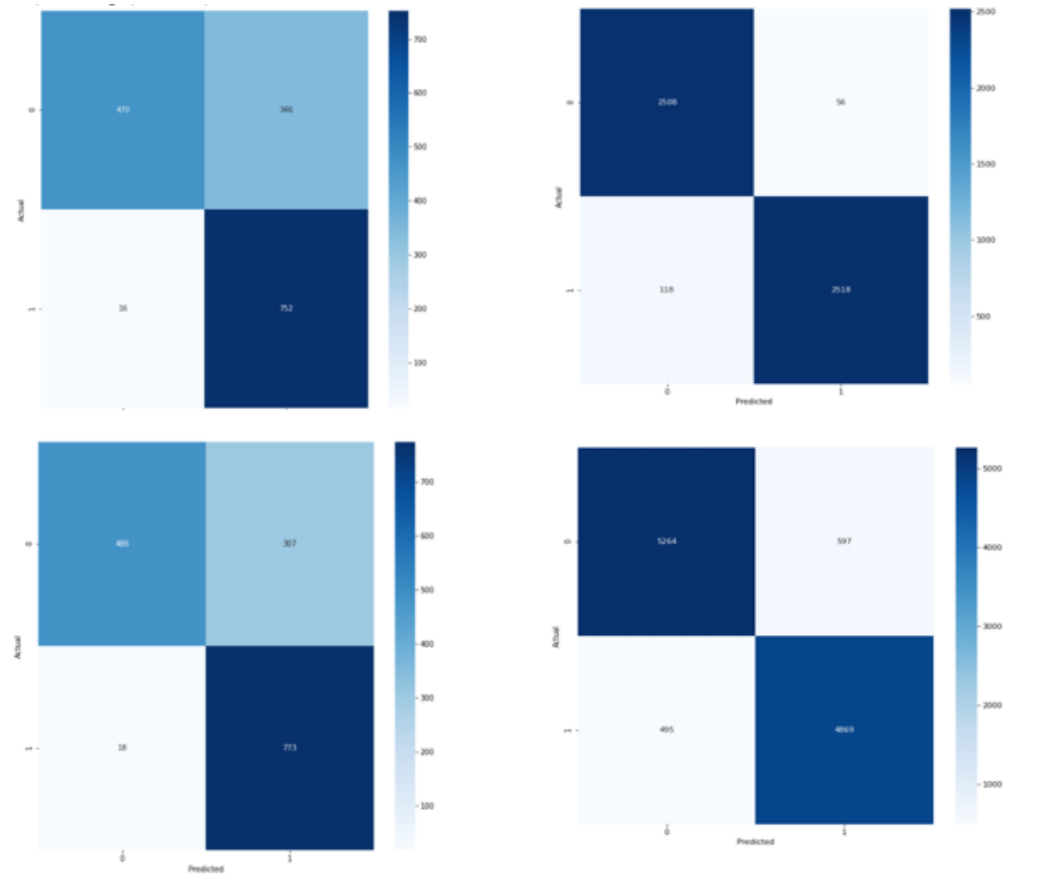


FIGURE 3.3 – Matrice de confusion pour chaque ensemble de données du modèle CNN

Le modèle Naive Bayes (avec n-gram) a montré les meilleures performances parmi les modèles traditionnels d'apprentissage automatique, tandis que CNN, BiLSTM et Conv-HAN sont les plus prometteurs parmi les modèles basés sur les réseaux de neurones (tableau). Nous pouvons voir que les caractéristiques n-gram sont très prometteuses dans la détection de spam [208]. Par conséquent, la performance exceptionnelle des caractéristiques n-gram dans la détection des fausses nouvelles n'est pas surprenante. Nous constatons que les performances du modèle Naive Bayes (avec n-gram) sont presque équivalentes aux performances de ces modèles basés sur les réseaux de neurones. Par conséquent, Naive Bayes avec n-gram est notre modèle recommandé ajoutant la caractéristique NLP. Le modèle CNN unidimensionnel maintient des performances modérées sur tous les ensembles de données. Par conséquent, les modèles basés sur les réseaux de neurones peuvent montrer de hautes performances sur un plus grand ensemble de données, mais dans d'autres ensembles de données, ces modèles seront vulnérables au surapprentissage même si leurs performances sont élevées. D'autre part, le modèle hybride proposé [207] Conv-HAN montre de hautes performances sur seulement 2 ensembles de données, contrairement à ce qui est mentionné dans [207]. Naive Bayes est un bon choix qui mérite une exploration future avec un ensemble de données plus large.

Conclusion

Ce chapitre a présenté une analyse détaillée des ensembles de données, des techniques d'optimisation et des métriques d'évaluation utilisées dans le cadre de la détection des fausses nouvelles, ainsi qu'une étude comparative approfondie des différentes architectures de modèles de classification. Nous avons identifier les techniques et les modèles les plus efficaces pour la détection des fausses nouvelles. Ce chapitre prépare le terrain pour une analyse plus approfondie des performances des modèles dans des contextes d'application réels, tout en soulignant l'importance d'une optimisation adaptée et d'une évaluation rigoureuse.

Chapitre 4

Détection des Fausses Nouvelles Textuelles : Contributions Basées sur le Deep Learning et le Traitement du Langage Naturel

Introduction

Dans les chapitres précédents, nous avons exploré les concepts fondamentaux de la détection des fausses nouvelles, en nous concentrant sur les différentes approches textuelles et les processus d'apprentissage automatique. Nous avons également présenté les ensembles de données et les métriques d'évaluation qui jouent un rôle clé dans la validation des performances des modèles. À présent, nous nous tournons vers une contribution originale à ce domaine en nous appuyant sur les modèles d'apprentissage profond pour améliorer la détection des fausses nouvelles.

Ce chapitre se concentre sur la proposition et l'évaluation de différentes approches basées sur l'intelligence artificielle pour résoudre le problème de la désinformation. Tout d'abord, nous offrons un aperçu des techniques d'apprentissage profond utilisées pour détecter les fausses nouvelles, en analysant leur efficacité dans divers contextes expérimentaux. Cette section couvre les informations statistiques sur les ensembles de données et les caractéristiques textuelles étudiées, ainsi que les modèles implémentés et leur processus d'évaluation. Ensuite, nous présenterons une amélioration spécifique de la détection des fausses nouvelles en combinant des modèles LSTM et BiLSTM avec des techniques d'embedding de mots. Cette section mettra en lumière le prétraitement NLP, l'extraction des caractéristiques et les résultats obtenus. Par ailleurs, une analyse comparative sur la détection du spam utilisant des modèles d'apprentissage automatique et d'apprentissage profond sera discutée. Cette approche se base sur l'extraction des caractéristiques NLP, et vise à montrer comment des techniques similaires peuvent être appliquées à des problèmes connexes de classification textuelle.Enfin, nous proposerons une étude appliquée au contexte spécifique du tremblement de terre au Maroc, dans laquelle une approche basée sur l'intelligence artificielle est mise en œuvre pour la détection des fausses nouvelles dans des scénarios de crise. Cette partie se concentrera sur une comparaison des performances entre les modèles CNN, BiLSTM, et HAN, tout en discutant des forces et limites de l'étude.

4.1 Intelligence Artificielle pour les Fausses Nouvelles

4.1.1 Aperçu de l'approche

Dans notre cadre proposé, illustré à la Fig.4.1, nous commençons par la collecte de données pour entraîner nos modèles, en prétraitant l'ensemble de données et en supprimant les caractères et mots inutiles. Les entités N-gram sont extraites et une matrice d'entités est formée pour représenter les documents concernés. La dernière étape du processus de classification consiste à entraîner le classificateur. Nous avons étudié différents classificateurs pour prédire la classe des documents. Nous avons spécifiquement étudié 6 algorithmes d'apprentissage profond différents : CNN, LSTM, BI-LSTM, HAN, HAN Convolutionnel et Bert.

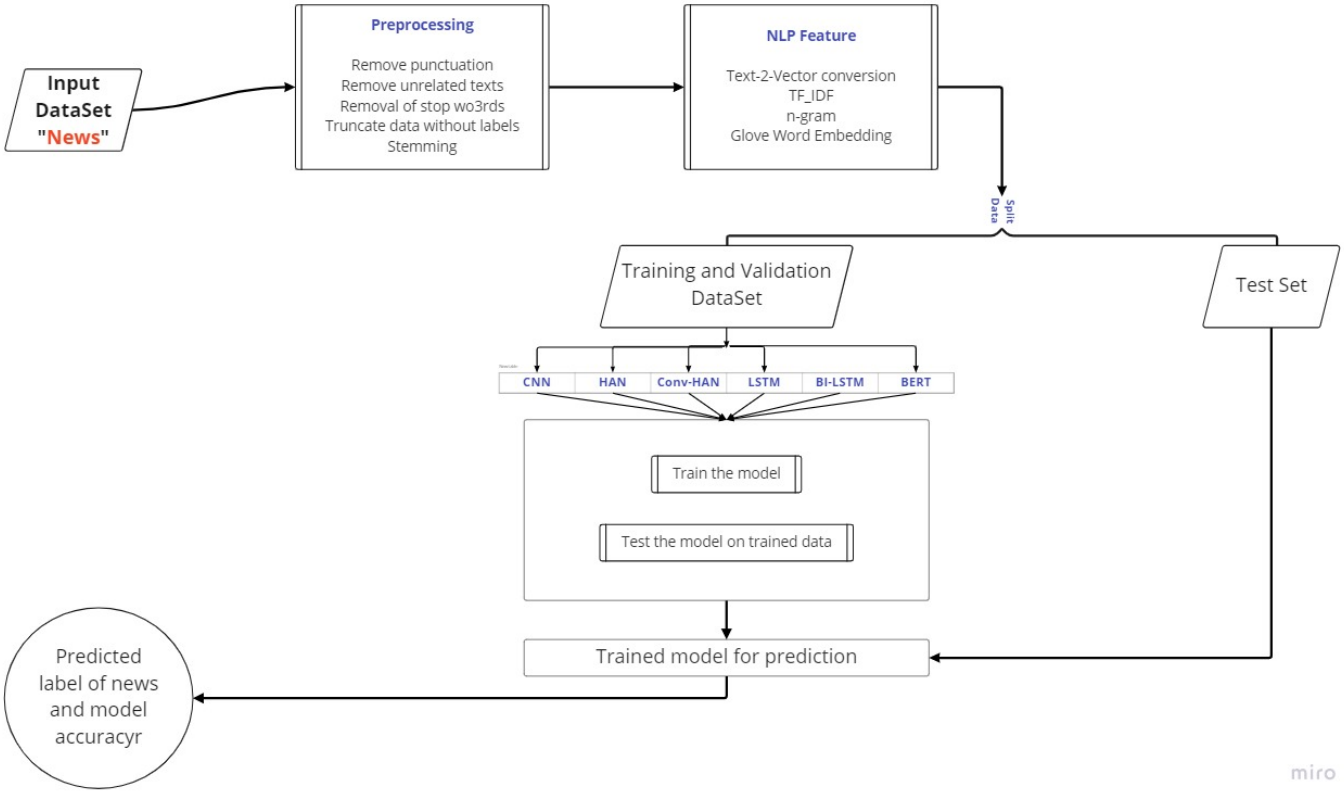


FIGURE 4.1 – Flux de travail pour l'entraînement des algorithmes et la classification des nouvelles.[8]

4.1.2 Évaluation Expérimentale

Informations Statistiques sur les Ensembles de Données

L'ensemble de données ISOT Fake News, créé par le laboratoire de recherche ISOT à l'Université de Victoria au Canada [36], est le plus grand ensemble de données d'histoires complètes de fausses

Type de Nouvelles	Réel	Faux
Catégorie	Nouvelles mondiales Nouvelles politiques	Nouvelles gouvernementales Nouvelles du Moyen-Orient Nouvelles des États-Unis Nouvelles du parti de gauche Nouvelles politiques Nouvelles générales
Taille	21 417	23 481

TABLE 4.1 – Les fragments de l'ensemble de données ISOT [8].

nouvelles accessible. Il y a 44 898 éléments dans la collection ISOT, avec 21 417 classés comme Réels et 23 481 désignés comme Faux (voir Tab.4.1).

Chaque modèle sera d'abord entraîné avec 80% des données ISOT. Les 20% restants des données ISOT seront utilisés pour évaluer la précision des classificateurs entraînés. FakeNewsNet et les Données Originales seront également utilisés pour les tests, comme mentionné précédemment. Le but de ces tests supplémentaires est de s'assurer que nous détectons les fausses nouvelles plutôt que tout autre motif dans l'ensemble de données ISOT, tel que le style d'une organisation de presse.

Chaque article de l'ensemble de données ISOT étiqueté comme Réel a été obtenu de Reuters ; toutes les histoires commençaient par le terme "Reuters". Les humains et les robots pourraient facilement reconnaître ce motif. Pour éviter cela, le terme "Reuters" a été omis au début de chaque élément.

Nous utilisons des nuages de mots pour les nouvelles réelles et fausses afin de visualiser cet ensemble de données et d'en obtenir un aperçu. La Fig. 4.2 montre un nuage de mots des nouvelles réelles et fausses de l'ensemble de données.

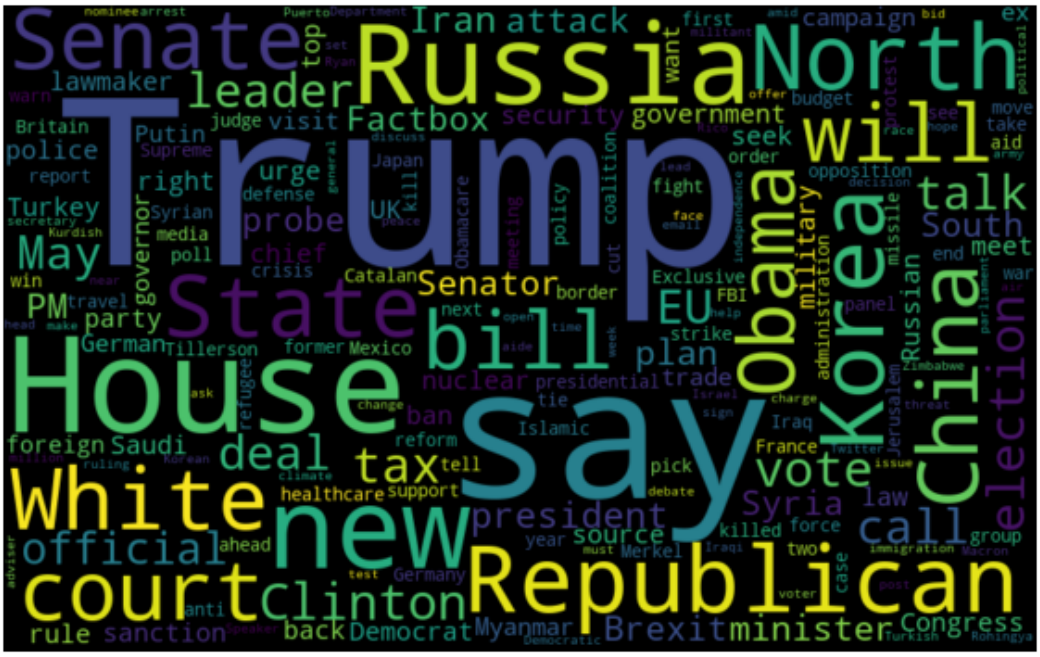


FIGURE 4.2 – Nuage de mots des ensembles de données réelles et fausses [8]

Caractéristiques Étudiées

Tf-Idf proposée par [209] et largement utilisée dans de nombreuses tâches en traitement du langage naturel. La méthode TF-IDF permet de quantifier les mots en reflétant l'importance d'un mot pour un document dans un corpus de documents. Cette méthode repose sur l'idée que chaque mot reçoit un poids propre w_{ij} basé sur son apparition dans le document et dans l'ensemble des documents. Ces poids mettent en avant les mots qui sont distincts et contiennent des informations utiles dans un document donné. "Ainsi, l'idf d'un terme rare est élevé, tandis que l'idf d'un terme fréquent est susceptible d'être faible" [210]. Pour chaque mot dans un document j, la valeur TF-IDF est calculée en d'abord calculant la Fréquence du Terme (TF), qui compte le nombre d'occurrences des mots dans un document, puis la Fréquence Inverse du Document (IDF), qui est le catalyseur pour s'assurer que les mots apparaissant moins fréquemment reçoivent plus de poids par rapport à ceux apparaissant plus fréquemment (par exemple, les mots vides), ce qui est calculé comme suit :

$$\log\left(\frac{|D|}{d_{fi}}\right)$$

où d_{fi} désigne le nombre de documents contenant le mot i et $|D|$ se réfère au nombre de documents dans le corpus. La métrique TF-IDF est calculée comme suit :

$$w_i = t_{fi} \cdot \log\left(\frac{|D|}{d_{fi}}\right)$$

où t_{fi}, d_{fi}, et $|D|$ se réfèrent respectivement au nombre d'apparitions du mot i dans le document j, au nombre de documents contenant le mot i, et au nombre total de documents. Cependant, cette méthode ne permet pas de capturer les motifs sémantiques, ce qui la rend utile uniquement pour les caractéristiques lexicales.

Intégration de mots (word embedding) L'intégration de mots est un sous-ensemble de techniques de représentation vectorielle dense pour représenter les mots et les textes.Les méthodes traditionnelles de codage de motifs de sac de mots employaient de grands vecteurs dispersés pour représenter chaque mot ou pour faire une marque dans un vecteur pour représenter un vocabulaire complet. Deux exemples de méthodes pour apprendre à intégrer des mots à partir de textes : Word2Vec et GloVe. En plus de ces méthodes soigneusement conçues, l'intégration de mots peut être apprise dans le cadre d'un modèle d'apprentissage profond. Cela peut être une approche plus lente, mais elle permet d'adapter le modèle à un ensemble de données d'entraînement spécifique.

Word2Vec est disponible en deux modes : sac de mots continu (CBOW) et skip-gram. Il a été initialement conçu pour prédire un mot dans un contexte. Par exemple, étant donné deux mots précédents et les deux mots suivants, quel mot est le plus susceptible de se produire entre eux. Mais il semble que la représentation cachée de ces mots fonctionne bien comme intégration de mots et possède des propriétés très intéressantes telles que les mots ayant un sens similaire ont une représentation vectorielle similaire.

Il est également possible d'effectuer des calculs arithmétiques qui capturent des informations telles que le singulier, le pluriel ou même les capitales et les pays.

GloVe est une technique d'apprentissage non supervisée célèbre pour la représentation des mots dans l'espace vectoriel, développée par Stanford en 2014 [114]. Elle tire parti du modèle skip-gram de word2vec et des méthodes de filtrage collaboratif, également connues sous le nom de factorisation matricielle. GloVe crée simplement une matrice de co-occurrence de mots à partir de l'ensemble du document utilisé pour l'entraînement et mappe chaque mot à un endroit sémantiquement pertinent dans l'espace, en gardant la distance entre les mots liés minimale [85].

Extraction des Caractéristiques dans le Modèle

Prétraitement des Données Les données textuelles nécessitent un prétraitement spécial pour être mises en œuvre dans des algorithmes d'apprentissage automatique ou d'apprentissage profond. Il existe diverses techniques largement utilisées pour convertir les données textuelles en une forme prête pour la modélisation. Les étapes de prétraitement des données que nous décrivons ci-dessous sont appliquées au contenu des nouvelles. Les différentes représentations de vecteurs de mots que nous avons employées dans le cadre de notre analyse sont également données.

Nuage de mots : Avant de commencer le prétraitement, nous visualisons nos données à partir du nuage de mots des mots-clés les plus utilisés dans nos données.

Suppression de la Ponctuation : La ponctuation dans le langage naturel fournit le contexte grammatical de la phrase. Les signes de ponctuation tels qu'une virgule peuvent ne pas ajouter beaucoup de valeur à la compréhension du sens de la phrase.

Suppression des mots vides : Nous commençons par supprimer les mots vides des données textuelles disponibles. Les mots vides sont des mots insignifiants dans une langue qui créeront du bruit lorsqu'ils seront utilisés comme caractéristiques dans la classification de texte. Les mots les plus courants dans une langue qui ne fournissent pas beaucoup de contexte peuvent être traités et filtrés du texte car ils sont plus courants et contiennent moins d'informations utiles. Nous avons utilisé la bibliothèque Natural Language Toolkit (NLTK) pour supprimer les mots vides.

Racination : La racination est une technique de suppression des préfixes et des suffixes d'un mot, se terminant par la racine. En utilisant la racine, nous pouvons réduire les formes fléchies et parfois les formes dérivées d'un mot à une forme de base commune.

Extraction des Caractéristiques La performance des modèles d'apprentissage profond dépend en grande partie de la conception des caractéristiques.

Extraction des caractéristiques n-grammes : Le n-gramme basé sur les mots a été utilisé pour représenter le contexte du document et générer des fonctionnalités pour classer le document comme faux et réel. De nombreux travaux existants ont utilisé les approches unigramme (n = 1) et bigramme (n = 2) pour la détection des fausses nouvelles [211]. Nous avons utilisé la fonction TfidfVectorizer de la

bibliothèque sklearn d'extraction de caractéristiques en Python pour générer des fonctionnalités TF-IDF n-grammes.

Intégration de Mots Pré-entraînée : Pour les modèles de réseaux de neurones, les intégrations de mots ont été initialisées avec des intégrations pré-entraînées de GloVe en 100 dimensions [36]. Leur entraînement s'est déroulé sur un ensemble de données comprenant un milliard de tokens (mots) et un vocabulaire de 400 000 mots. Dans cette architecture LSTM avec gensim, nous avons utilisé Word2Vec de Google pour représenter les mots dans des intégrations d'espace vectoriel de 100 dimensions.

Sac de Mots : La technique du sac de mots traite chaque article de presse comme un document et calcule le nombre de fréquences de chaque mot dans ce document, qui est ensuite utilisé pour créer une représentation numérique des données, également connue sous le nom de caractéristiques vectorielles de longueur fixe. Le sac de mots convertit le texte brut en vecteur de comptage des mots avec la fonction CountVectorizer pour l'extraction des caractéristiques. CountVectorizer divise le contenu du texte, construit le vocabulaire, et encode le texte en un vecteur. Ce vecteur encodé aura un comptage des occurrences de chaque mot qui ressemble davantage à un comptage de fréquence sous forme de paire clé/valeur. Cette méthodologie présente des inconvénients en termes de perte d'information. La position relative des mots est ignorée et l'information contextuelle est perdue.

Modèles Étudiés pour l'Implémentation des Approches

Ici, nous décrivons d'abord la configuration expérimentale des différents modèles basés sur les réseaux de neurones et l'apprentissage profond utilisés dans notre expérience.

CNN Le modèle de Réseaux de Neurones Convolutifs a été initialisé comme une séquence de couches. Nous utiliserons une structure de réseau entièrement connecté avec trois couches. Les couches entièrement connectées sont définies en utilisant la classe Dense. La configuration des couches neurales permet de définir le nombre d'unités de traitement et leur fonction d'activation. Pour les deux premières strates, nous optons pour l'activation ReLU (unité linéaire rectifiée), tandis que la couche finale emploie une fonction sigmoïde. Le choix de la sigmoïde en sortie garantit que les valeurs produites se situent dans l'intervalle [0, 1]. Cela facilite l'interprétation en termes de probabilité d'appartenance à la classe 1, ou permet une classification binaire stricte en appliquant un seuil de 0,5 par défaut. La compilation du modèle intègre l'algorithme d'optimisation ADAM, paramétré avec un taux d'apprentissage de 0,001, visant à réduire la perte d'entropie croisée binaire. L'entraînement du réseau s'est déroulé sur 3 itérations complètes du jeu de données.

LSTM **LSTM avec gensim** Nous utilisons la bibliothèque gensim en Python qui prend en charge une multitude de classes pour les applications NLP. Comme discuté, nous utilisons un modèle CBOW avec échantillonnage négatif et des vecteurs de mots de 100 dimensions. L'optimiseur ADAM avec un taux d'apprentissage de 0,001 a été appliqué pour minimiser la perte d'entropie croisée binaire et la sigmoïde

était la fonction d'activation pour la couche de sortie finale. Enfin, ce modèle a été entraîné sur 3 époques avec des lots de 64 et 512.

LSTM avec GloVe Le modèle LSTM a été pré-entraîné avec des intégrations GloVe en 100 dimensions. La dimension de sortie et les pas de temps ont été réglés à 300. L'optimiseur ADAM avec un taux d'apprentissage de 0,001 a été appliqué pour minimiser la perte d'entropie croisée binaire et la sigmoïde était la fonction d'activation pour la couche de sortie finale. Enfin, ce modèle a été entraîné sur 3 époques avec des lots de 64 et 512.

Bi-LSTM Le but du modèle Bi-LSTM est de détecter les anomalies dans une certaine partie des nouvelles, nous devons l'examiner avec les événements d'action précédents et suivants. Le Bi-LSTM a été initialisé avec des intégrations GloVe pré-entraînées de 100 dimensions. Une dimension de sortie de 100 et des pas de temps de 300 ont été appliqués. L'optimiseur ADAM avec un taux d'apprentissage de 0,001 a été utilisé pour minimiser la perte d'entropie croisée binaire. La taille du lot d'apprentissage a été fixée à 128 et une perte à chaque époque a été observée avec rappel.

HAN Le réseau d'attention hiérarchique (HAN) consistait en deux mécanismes d'attention pour le codage au niveau des mots et des phrases. Avant l'entraînement, nous avons fixé le nombre maximum de phrases dans un article de presse à 20 et le nombre maximum de mots dans une phrase à 100. Dans les deux niveaux de codage, un GRU bidirectionnel avec une dimension de sortie de 100 a été introduit dans notre couche d'attention personnalisée. Nous avons utilisé un encodeur de mots comme entrée de notre couche d'encodeur de phrases distribuée dans le temps. Nous avons optimisé notre modèle avec ADAM qui a appris à un taux de 0,001. Nous avons utilisé la bibliothèque Keras avec le backend Tensorflow pour implémenter le mécanisme d'attention.

HAN Convolutionnel Afin d'extraire des caractéristiques d'entrée de haut niveau, nous avons incorporé une couche convolutive unidimensionnelle avant chaque couche GRU bidirectionnelle dans le HAN. Cette couche a sélectionné les caractéristiques de chaque trigramme de l'article de presse avant de les transmettre à la couche d'attention.

BERT Les Représentations Encodeur Bidirectionnel de Transformers (BERT) sont un modèle pré-entraîné pour l'apprentissage des représentations contextuelles des mots à partir de textes non étiquetés. Il possède deux caractéristiques principales : c'est un modèle de transformateur profond qui peut analyser efficacement de longues phrases en utilisant le mécanisme d'« attention », et il est bidirectionnel, ce qui signifie qu'il produit des sorties basées sur l'ensemble de la phrase d'entrée. Nous avons utilisé BERT pour gérer l'ensemble de données et construire un modèle d'apprentissage profond pour la détection des fausses nouvelles en ajustant finement le modèle pré-entraîné bert-based-uncased. Comme le modèle BERT-Large nécessite beaucoup de temps et de mémoire, nous avons choisi BERT-Base pour cette enquête. Le modèle BERT-Base est composé de 12 couches (blocs de transformateurs) et de 110 millions de paramètres.

4.1.3 Résultats

Dans cette section, nous décrivons une analyse des performances de nos modèles basés sur les réseaux de neurones et l'apprentissage profond. Nous présentons les meilleures performances pour chaque ensemble de données. Nous calculons la précision, la précision, le rappel et le score F1 pour les classes fausses et réelles, et trouvons leur moyenne, moyenne pondérée (le nombre d'instances vraies pour chaque classe) et rapportons un score moyen de ces métriques.

Métriques d'Évaluation

Nous utilisons la précision, la précision, le rappel et le score F1 comme métriques d'évaluation (tp, fp, fn dans les équations suivantes sont respectivement les vrais positifs, faux positifs et faux négatifs). La précision est une mesure calculée comme le ratio des prédictions correctes au nombre total d'exemples. La précision mesure le pourcentage de prédictions positives qui sont correctes et est définie comme suit :

$$Prcision = \frac{tp}{tp + fp} \tag{4.1}$$

Le rappel consiste à mesurer le pourcentage de prédictions correctes que le classificateur capture et est défini comme suit :

$$Rappel = \frac{tp}{tp + fn} \tag{4.2}$$

Le score F1 permet de trouver l'équilibre entre le rappel et la précision et est calculé comme suit :

$$ScoreF1 = \frac{\text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{4.3}$$

La précision est souvent la métrique la plus utilisée représentant le pourcentage d'observations correctement prédites, qu'elles soient vraies ou fausses. Pour calculer la précision des performances d'un modèle, l'équation suivante peut être utilisée :

$$Prcision = \frac{tp + tn}{tp + tn + fp + fn} \tag{4.4}$$

Modèles	Caractéristiques	Exactitude	Précision	Rappel	Score F1
LSTM	word2vec (Gensim)	0.98	0.98	0.98	0.98
LSTM	Glove Embedding	0.98	0.98	0.98	0.98
BiLSTM		0.78	0.50	0.78	0.89
HAN		0.59	0.59	0.59	0.59
Conv-HAN		0.56	0.56	0.56	0.56
CNN	TF-idf Countvectorizer	0.90	0.90	0.90	0.90
BERT		0.99	0.99	0.98	0.99

TABLE 4.2 – Résultats des modèles prédictifs sur les quatre ensembles de données

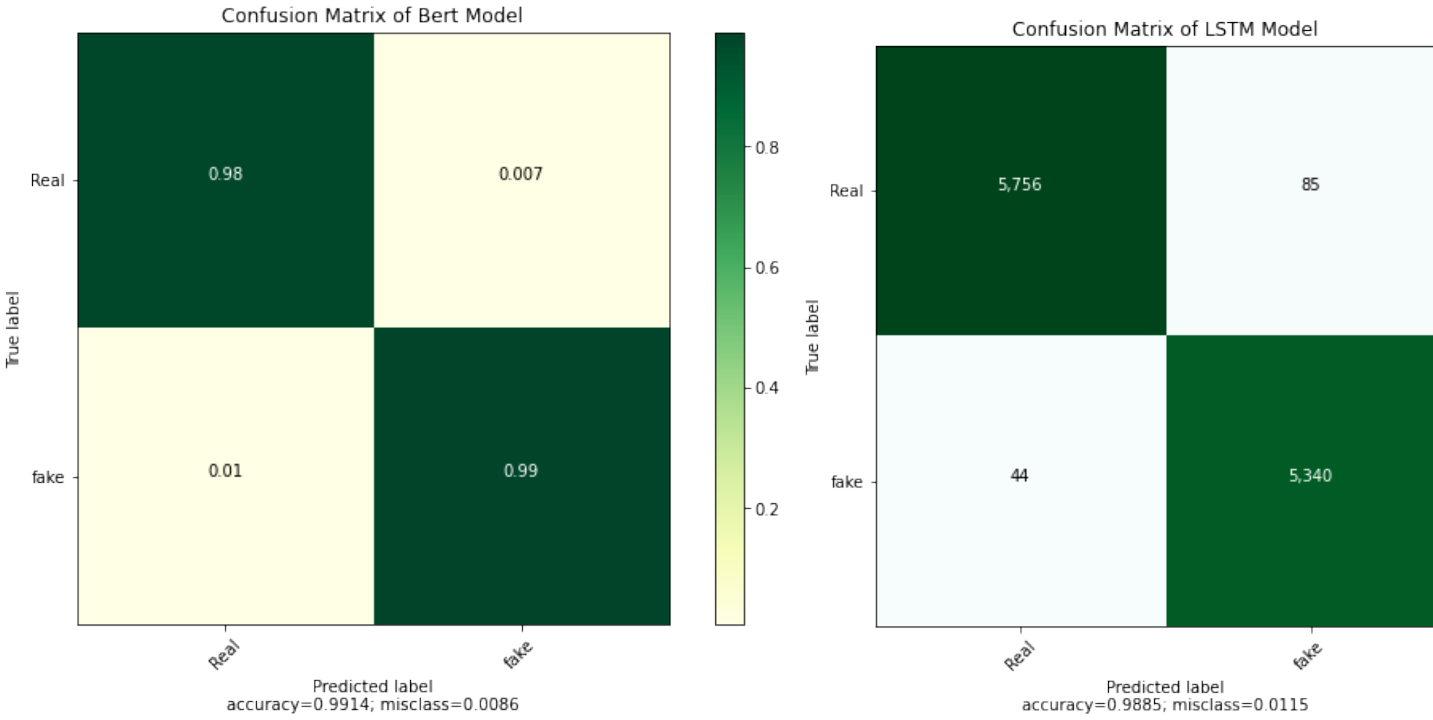


FIGURE 4.3 – Matrice de confusion pour les modèles Bert et LSTM

4.1.4 Résultats et discussion

Les études précédentes sur la détection des fausses nouvelles se sont principalement concentrées sur les modèles traditionnels d'apprentissage automatique. Il est donc important de comparer leurs performances avec celles des modèles d'apprentissage profond. En particulier, l'objectif de l'étude précédente est de comparer les performances de différents modèles traditionnels d'apprentissage automatique et de modèles d'apprentissage profond sur la détection des fausses nouvelles. Compte tenu du grand succès des modèles de langage avancés pré-entraînés sur diverses tâches de classification de texte, dans le Tab. 4.3, nous rapportons les performances de différents modèles d'apprentissage profond. Le modèle de base CNN est considéré comme le meilleur modèle pour Liar dans [163]. Nous constatons qu'il est le troisième meilleur modèle basé sur les réseaux de neurones selon ses performances sur l'ensemble de données. Les modèles basés sur BILSTM sont les plus vulnérables au surapprentissage pour cet ensemble de données, ce qui se reflète dans leurs performances. Bien que HAN soit également victime de surapprentissage, comme mentionné dans [163], Notre LSTM avec gensim ou avec GloVe présente les meilleures performances parmi les modèles neuronaux pour l'ensemble de données, avec une précision de 98% et un score F1 de 0,98. Les modèles CNN montrent une amélioration sur l'ensemble de données, tandis que les modèles CNN continuent leurs performances impressionnantes 4.5. Parmi les modèles d'apprentissage profond en traitement du langage naturel avancés pré-entraînés que nous avons étudiés, Bert montre les meilleures performances avec une précision de 0,99. Les modèles basés sur BERT pré-entraînés surpassent les autres modèles. Nous voyons que le modèle basé sur BERT est capable d'atteindre une haute précision (plus de 90%) Par conséquent, ces modèles peuvent être utilisés pour la détection des fausses

nouvelles dans différentes langues où une grande collection de données étiquetées n'est pas réalisable. Différents modèles BERT pré-entraînés sont déjà disponibles pour différentes langues.

4.2 Amélioration de la détection de la désinformation en utilisant LSTM et BiLSTM avec des techniques d'embedding de mots

4.2.1 Modèle proposé pour la détection des fausses nouvelles

Le modèle proposé pour la détection des fausses nouvelles implique une approche systématique et structurée, en utilisant des techniques avancées de NLP et d'apprentissage automatique. Le flux de travail est résumé visuellement dans la Figure 4.4 et est détaillé comme suit :

Jeu de données d'entrée

Le modèle commence par le jeu de données d'entrée ISOT dataset [], qui se compose d'articles de presse. Ce jeu de données est un mélange de vraies et fausses nouvelles, servant de base pour l'entraînement et l'évaluation des algorithmes de détection. Le jeu de données sera ensuite divisé en deux parties :

Jeu de données d'entraînement et de validation : 80% du jeu de données est utilisé pour entraîner les modèles et valider leurs performances. Jeu de test : Les 20% restants sont mis de côté pour tester les performances finales du modèle.

Prétraitement NLP

Pour exécuter des algorithmes d'apprentissage automatique ou d'apprentissage profond sur des données textuelles, un prétraitement significatif est nécessaire. Les données textuelles peuvent être transformées à l'aide de diverses méthodes pour créer une forme adaptée à la modélisation [212]. Les méthodes de prétraitement des données que nous décrivons ci-dessous sont appliquées au contenu des nouvelles. Nous fournissons également des détails sur les différentes représentations de vecteurs de mots que nous avons utilisées au cours de notre investigation.

Suppression des mots vides : Ensuite, nous éliminons les mots vides des données textuelles disponibles. Les mots vides sont de petits mots insignifiants qui, lorsqu'ils sont utilisés comme caractéristiques de classification de texte, seront perturbateurs pour le processus. Parce qu'ils sont plus courants et contiennent moins d'informations pertinentes, les mots les plus couramment utilisés dans une langue peuvent être traités et filtrés du texte. Les mots vides ont été éliminés en utilisant la bibliothèque Natural Language Toolkit (NLTK).

Suppression de la ponctuation : La ponctuation dans le langage naturel fournit le contexte grammatical de la phrase. Les virgules et autres signes de ponctuation peuvent ne pas contribuer de manière significative à la compréhension du contenu d'une phrase.

Racinisation : est une méthode pour supprimer les préfixes et les suffixes d'un mot, ne laissant que la racine. Les formes fléchies et parfois les formes dérivées d'un mot peuvent être réduites à une forme de base en utilisant la racine.

Extraction des caractéristiques

Pour construire un modèle d'apprentissage profond, la sélection des caractéristiques est d'une importance capitale pour une performance optimale du système. L'extraction des caractéristiques est une étape cruciale dans le traitement du langage naturel pour la détection de la désinformation. L'objectif de l'extraction des caractéristiques est de convertir les données textuelles brutes en un ensemble de caractéristiques numériques pouvant être utilisées comme entrée pour les algorithmes d'apprentissage automatique, tels que les réseaux neuronaux profonds.

Il existe de nombreuses approches différentes pour l'extraction des caractéristiques pour la détection de la désinformation, mais certaines techniques couramment utilisées incluent :

Sac de mots : Cette méthode simple compte la fréquence de chaque mot dans le texte en présentant les données textuelles comme une collection de mots. En conséquence, un vecteur clairsemé est créé, chaque membre indiquant la fréquence d'un mot différent dans l'entrée de texte. Bien que BoW soit simple à calculer, il ne capture pas le contexte ni l'ordre des mots.

Term Frequency-Inverse Document Frequency : Il s'agit d'un développement de BoW qui pondère l'importance de chaque mot dans le texte en fonction de sa fréquence d'apparition dans le document individuel et dans le corpus dans son ensemble [213]. Chaque mot est pondéré par TF-IDF de manière inversement proportionnelle à sa fréquence dans l'ensemble du corpus et proportionnelle à sa fréquence d'apparition dans le document. Cette approche améliore la différenciation entre l'information authentique et la désinformation en soulignant les expressions essentielles.

Vecteurs de mots : Dans cette méthode plus avancée, les mots sont représentés sous forme de vecteurs denses dans un espace de haute dimension, où la distance entre les vecteurs indique à quel point les mots sont sémantiquement proches les uns des autres. Les intégrations de mots sont souvent découvertes à l'aide de réseaux neuronaux entraînés sur de vastes bibliothèques de données textuelles, comme Word2Vec ou GloVe. L'avantage des intégrations de mots est qu'elles peuvent capturer les relations sémantiques entre les mots, mais leur calcul peut être coûteux en termes de ressources [213].

GloVe a été créé par Stanford en 2014 [114], une méthode d'apprentissage non supervisée bien connue pour la représentation des mots dans l'espace vectoriel. Elle utilise la factorisation matricielle, également connue sous le nom de filtrage collaboratif, et le modèle skip-gram de Word2Vec. Pour minimiser la distance entre les mots apparentés, GloVe extrait simplement une matrice de cooccurrence de mots à partir de l'ensemble du document d'entraînement et mappe chaque mot à un emplacement sémantiquement approprié dans l'espace .

Globalement, l'extraction des caractéristiques est une étape cruciale dans le NLP pour la détection de la désinformation, et il existe une variété de stratégies pouvant être appliquées en fonction des particularités de la tâche.

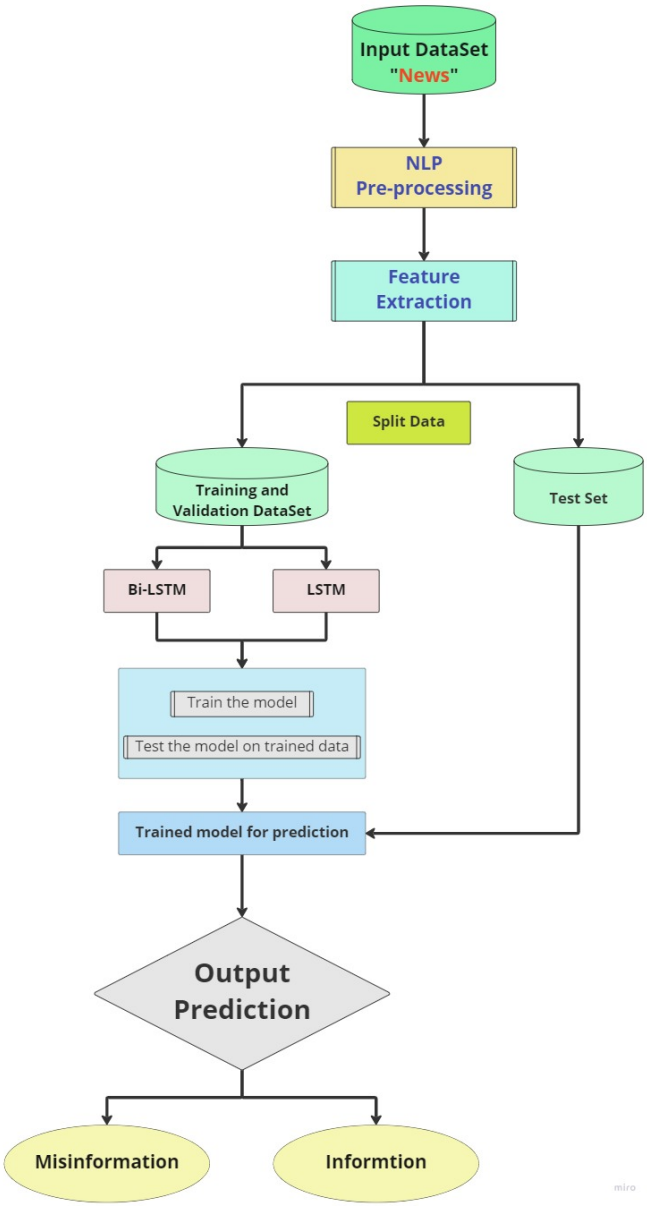


FIGURE 4.4 – Système de detection des fausses informations s'appuyant sur les architectures neuronales LSTM et Bi-directional LSTM

4.2.2 Résultats

Dans cette section, nous décrivons une analyse des performances de nos modèles basés sur les réseaux de neurones et l'apprentissage profond. Nous présentons les meilleures performances pour chaque ensemble de données. Nous calculons la précision, la précision, le rappel et le score F1 pour les classes fausses et réelles, et trouvons leur moyenne, moyenne pondérée (le nombre d'instances vraies pour chaque classe) et rapportons un score moyen de ces métriques.

Métriques d'Évaluation

Nous utilisons la précision, la précision, le rappel et le score F1 comme métriques de notation (tp, fp, fn dans les équations ci-dessous sont respectivement les vrais positifs, les faux positifs et les faux négatifs). La précision est une mesure calculée comme le ratio des prédictions correctes au nombre total d'échantillons. La précision mesure le pourcentage de prédictions positives qui sont correctes et est définie comme suit :

$$Prcision = \frac{tp}{tp + fp} \tag{4.5}$$

Le rappel consiste à mesurer le pourcentage de prédictions correctes que le classificateur capture et est défini comme suit :

$$Rappel = \frac{tp}{tp + fn} \tag{4.6}$$

Le score F1 vise à trouver un équilibre entre le rappel et la précision et est calculé comme suit :

$$ScoreF1 = \frac{\text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{4.7}$$

La précision est souvent la métrique la plus couramment utilisée et représente le pourcentage d'observations correctement prédites (vraies ou fausses). Pour calculer la précision des performances de votre modèle, vous pouvez utiliser la formule suivante :

$$Prcision = \frac{tp + tn}{tp + tn + fp + fn} \tag{4.8}$$

Modèles	Caractéristiques	Précision	Rappel	F1-score
LSTM	word2vec (Gensim)	0,78	0,78	0,78
LSTM	GloVe Embedding	0,96	0,96	0,96
BiLSTM	GloVe Embedding	0,94	0,94	0,94

TABLE 4.3 – Résultats des modèles prédictifs LSTM et BI-LSTM sur les ensembles de données ISOT

Nous décrivons ici d'abord la configuration expérimentale pour divers modèles basés sur les réseaux neuronaux :

LSTM avec gensim : Le réseau et l'apprentissage profond utilisés dans l'expérience. LSTM avec Gensim nous utilisons leur bibliothèque Gensim pour Python. Un ensemble de classes pour les applications NLP. Utilisez le modèle CBOW comme décrit. Échantillonnage négatif et vecteurs de mots de 100 dimensions. L'optimiseur ADAM avec un taux d'apprentissage de 0,001 a été appliqué pour minimiser la perte d'entropie croisée binaire, Sigmoïde était la fonction d'activation pour la couche de sortie finale. Enfin, ce modèle a été entraîné sur 3 époques avec des lots de 64 et 512.

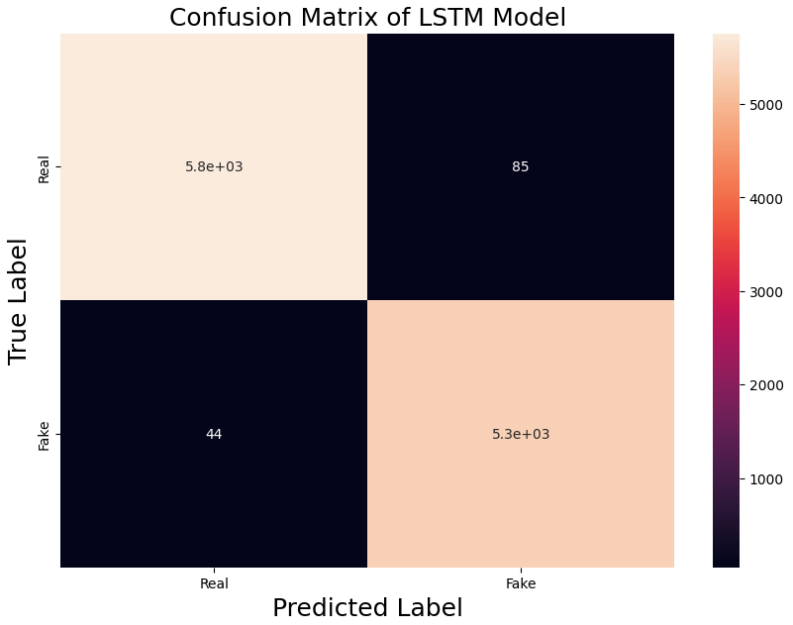


FIGURE 4.5 – Matrice de confusion pour le modèle LSTM

Modèle LSTM avec GloVe : Le modèle LSTM est pré-entraîné avec des intégrations GloVe de 100 dimensions. Les dimensions de sortie et le pas de temps sont réglés à 300. Un optimiseur ADAM avec un taux d'apprentissage de 0,001 a été appliqué pour minimiser la perte d'entropie croisée binaire, et la sigmoïde était la fonction d'activation de la couche de sortie finale. Enfin, le modèle a été entraîné en lots de 64 et 512 pendant 3 époques.

Bi-LSTM : Le but du modèle Bi-LSTM est de détecter les anomalies dans un modèle particulier. C'est une partie des nouvelles et elle doit être explorée à la fois avant et après les événements d'action. Le Bi-LSTM a été initialisé avec une intégration GloVe pré-entraînée de 100 dimensions. 100 dimensions de sortie et 300 pas de temps ont été appliqués. Nous avons utilisé l'optimiseur ADAM avec un taux d'apprentissage de 0,001 pour minimiser la perte d'entropie croisée binaire. La taille de la pile d'apprentissage a été fixée à 128 et nous avons observé une perte pour chaque époque sur le rappel.

Dans le Tab. 4.3, nous rapportons les performances de différents modèles d'apprentissage profond. Le succès supérieur des modèles Bi-LSTM dans la détection de la désinformation peut être dû à leur capacité à reconnaître des motifs et des connexions plus complexes dans les données. Les Bi-LSTM peuvent prendre en considération à la fois les mots qui précèdent et ceux qui suivent un autre mot dans une phrase en traitant la séquence d'entrée dans les deux directions, avant et arrière [214]. Cela peut les aider à comprendre le contexte global du texte. Cela est crucial pour détecter la désinformation parce qu'il peut être compliqué de distinguer les deux, même lorsque certains mots ou phrases sont inclus. En conclusion, les modèles Bi-LSTM ont démontré des performances supérieures dans l'identification des fausses nouvelles par rapport aux modèles LSTM avec une précision de 94% et un score F1 de 0,94. Ils sont un outil important dans la lutte contre la désinformation en raison de leur capacité à reconnaître des motifs et des connexions plus complexes dans les données. Le fait qu'il y ait toujours des possibilités

d'amélioration dans les algorithmes d'apprentissage automatique ne doit pas être négligé. Pour continuer à améliorer l'efficacité de ces modèles dans l'identification des fausses nouvelles et d'autres tâches de NLP, des recherches supplémentaires sont nécessaires.

4.3 Une analyse et évaluation de la détection de spam utilisant des techniques d'apprentissage automatique et d'apprentissage profond optimisées : application d'une nouvelle approche basée sur l'extraction des caractéristiques NLP

4.3.1 Modèle proposé

Notre cadre proposé pour la détection de SPAM peut être résumé par le schéma présenté dans la figure 4.6 . Comme illustré dans cette figure, le système suggéré comprend plusieurs phases, dont la première est l'acquisition de données, suivie par le prétraitement du corpus collecté. Ensuite, l'annotation a été réalisée manuellement. Après cela, la tokenisation, l'élimination des mots vides et le stemming ont été effectués. Enfin, notre modèle a été entraîné et testé avec les algorithmes choisis, à savoir K-Means, SVM et LSTM, en utilisant le corpus avec les techniques TF-IDF et word2vec. Des détails supplémentaires sur chaque phase sont fournis dans les sous-sections suivantes.

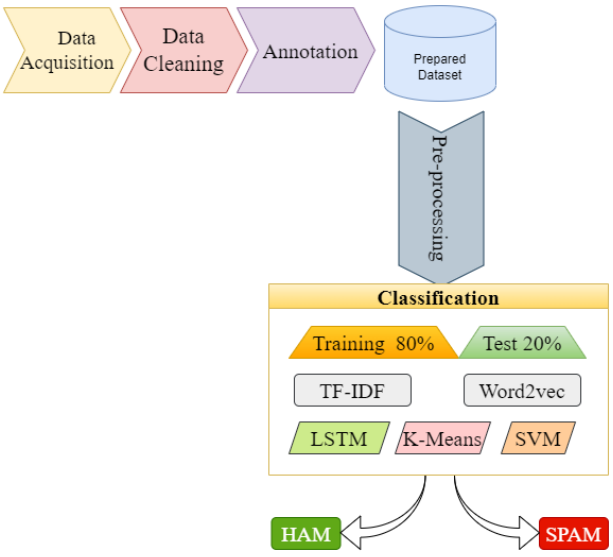


FIGURE 4.6 – L'architecture du modèle proposé

4.3.2 Description de l'ensemble de données

Le corpus de données employé pour notre étude provient de la plateforme Kaggle. Il contient environ 57 208 lignes. Comme indiqué dans le tableau 1, la caractéristique descriptive se compose du texte des

emails envoyés, tandis que la caractéristique ciblée se compose de deux classes : "Ham" et "Spam". Les classes sont étiquetées pour chaque message du jeu de données et représentent la fonctionnalité ciblée avec un alphabet binaire de type chaîne Ham ; Spam. Les classes sont ensuite mappées à l'entier 0 (Ham) et 1 (Spam).

TABLE 4.4 – Exemple de jeu de données utilisé

Message	**Catégorie**
Vous avez une annonce importante du service client de premier.	Spam
Vos crédits ont été rechargés pour `http://www.bubbletext.com` votre code de renouvellement est Txxxxx	Spam
La science dit que le chocolat fondra sous le soleil. Veuillez ne pas marcher sous le soleil. Baozi ne veut pas **perdre** un ami cher.	Ham
Le montant mensuel n'est pas si terrible et vous ne paierez rien avant 6 mois après avoir terminé l'école.	Ham

4.3.3 Prétraitement et Annotation

Afin de pouvoir créer et entraîner un classificateur à partir de n'importe quel corpus de texte, il est impératif que les documents textuels soient convertis en entrées valides qui peuvent être comprises par tout algorithme de classification, conformément à une procédure précise. En effet, ces entrées précieuses sont des vecteurs ou des matrices qui spécifient le poids de chaque descripteur (mot ou groupe de mots) pour chaque fichier texte où ils sont présents. Ainsi, le processus qui est effectué sur le corpus avant de le retourner sous forme de document vectoriel est connu sous le nom de prétraitement, tandis que la transformation d'un document textuel (non structuré) en document vectoriel (données structurées) est connue sous le nom d'extraction de caractéristiques. Par conséquent, les phases de prétraitement du corpus que nous suivons sont décrites dans la figure 4.7.

FIGURE 4.7 – Processus de prétraitement et de transformation du texte du courrier électronique

Tokenisation

La tokenisation est un processus de décomposition d'un fichier texte en une collection de mots et une collection de phrases. Son objectif est de calculer correctement les ensembles de données sous ces

formes, afin d'améliorer la précision des classificateurs. La décomposition du texte est le processus consistant à transformer un texte d'entrée en un ensemble de mots, y compris les répétitions.

Suppression des mots vides

Certains termes, désignés comme mots outils, se retrouvent abondamment dans tous types d'écrits. Leur omniprésence les rend peu significatifs pour catégoriser un texte. De plus, leur inclusion peut introduire des perturbations nuisant à l'efficacité du dispositif de classification. Par conséquent, il est recommandé d'éliminer ces éléments pour optimiser les capacités de catégorisation du système qui sera mis en œuvre ultérieurement. Cette catégorie englobe notamment les mots de liaison, les conjonctions, les articles, ainsi que certains verbes courants dans l'ensemble des classes de la typologie (tels que "il, votre, chapeau, avoir, être, donc, comment...").

Racine (Stemming)

En linguistique morphologique et dans le contexte de la recherche, l'extraction racinaire désigne le mécanisme de réduction des termes dérivés à leur forme élémentaire. La racine obtenue ne correspond pas systématiquement à la racine morphologique stricto sensu. L'objectif principal est de permettre le regroupement de mots partageant une base et une signification analogues, même si la racine extraite n'est pas forcément conforme aux critères linguistiques traditionnels.

Représentation vectorielle du texte

Un texte peut être perçu comme une suite de mots. Cependant, cette représentation n'est pas directement exploitable par les algorithmes d'apprentissage automatique, qui nécessitent des données sous forme de vecteurs numériques pour effectuer la classification. La vectorisation textuelle consiste à transformer chaque segment en une série de valeurs numériques, chacune correspondant à un terme du lexique établi à partir de l'ensemble des documents ou du corpus. Pour vectoriser les textes, on établit d'abord un inventaire exhaustif des termes présents dans le corpus d'entraînement. On génère ensuite une matrice numérique où chaque ligne représente un texte du corpus et chaque colonne un terme de l'inventaire. L'absence d'un terme dans un document est marquée par un 0, tandis que sa présence est indiquée soit par un 1, soit par le nombre total de ses occurrences dans le document. Cette structure matricielle est appelée matrice de fréquence.

Extraction des caracteristiques

La technique de pondération TF-IDF est fréquemment employée dans l'extraction d'informations, notamment en fouille de textes. Cet indicateur statistique évalue la pertinence d'un mot au sein d'un texte spécifique par rapport à un ensemble de documents ou un corpus. Sa magnitude reflète le degré d'indexation des termes dans la collection documentaire. Les composantes du vecteur représentant un

document correspondent aux valeurs attribuées à ses mots-clés d'indexation. On attribue un poids plus important aux termes caractérisant un document présenté comme

$$d = (w_1, w_2, w_3, \ldots, w_n) \tag{4.9}$$

Par conséquent, le TF-IDF est déterminé par l'équation 4.10.

$$\mathrm{TF}_i - \mathrm{IDF}_{i,j} = \mathrm{TF}_{i,j} * \mathrm{IDF}_i \tag{4.10}$$

Avec

$$\mathrm{TF}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{4.11}$$

Et

$$\mathrm{IDF}_i = \log\left(\frac{D}{d_{j:\, t_i \in d_j}}\right) \tag{4.12}$$

D'un autre côté, la méthode Word2vec, utilisée depuis 2013, est conçue pour créer efficacement des embeddings de mots. Cependant, au-delà de ses avantages en tant que méthode pour l'embedding des mots, de nombreux concepts associés à Word2vec se sont révélés efficaces pour créer des outils de recommandation et pour donner un sens aux ensembles de données, même pour des tâches non linguistiques et professionnelles.

Résultats et discussions

Une fois que nous avons préparé le corpus prétraité pour être utilisé par les classificateurs en utilisant les méthodes TF-IDF et Word2vec, nous commençons les phases d'apprentissage et d'évaluation. Pour cette dernière phase, une série de calculs de précision, d'exactitude et de f1-score est effectuée en utilisant les équations 4.13,4.14 et 4.15.

$$\text{Precision} = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Positive})} \tag{4.13}$$

$$\text{F1-Score} = 2 \times \frac{(\text{Sensitivity} \times \text{Precision})}{(\text{Sensitivity} + \text{Precision})} \tag{4.14}$$

$$\text{Accuracy} = \frac{(\text{True Positive} + \text{True Negative})}{(\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative})} \tag{4.15}$$

Les résultats de l'évaluation expérimentale de la classification en utilisant des classificateurs conventionnels, combinés avec les méthodes TF-IDF ou Word2vec, sont présentés dans le Tableau 4.5. À partir de ce tableau, on peut facilement observer que le classificateur LSTM atteint des valeurs de 98,01%,

TABLE 4.5 – Résultats des classificateurs incluant le prétraitement

Algorithm	**Technique**	**Precision**	**Accuracy**	**F1-score**
K-MEANS	TF-IDF	96.73%	95.90%	96.94%
	Word2vec	94.79%	92.54%	94.26%
LSTM	**TF-IDF**	**98.01%**	**97.79%**	**98.36%**
	Word2vec	94.84%	95.02%	96.24%
SVM	TF-IDF	91.03%	93.26%	93.39%
	Word2vec	90.36%	92.65%	92.91%

97,79% et 98,36% en précision, exactitude et F1-score, respectivement, pour notre corpus avec l'utilisation de TF-IDF. De plus, lorsque nous remplaçons TF-IDF par Word2vec, les valeurs obtenues sont de 94,84%, 95,02% et 96,24% en précision, exactitude et F1-score, respectivement. D'autre part, le classificateur SVM atteint 91,03% en précision, 93,26% en exactitude et 93,39% en F1-score lorsqu'on utilise TF-IDF. Cependant, le même algorithme atteint 90,36% en précision, 92,65% en exactitude, et 92,91% en F1-score avec l'utilisation de Word2vec. Lorsque TF-IDF est choisi, les résultats de K-means sont de 96,73%, 95,90% et 96,94% en précision, exactitude et F1-score, respectivement. De plus, avec l'utilisation de Word2vec, il atteint 94,79%, 92,54% et 94,26% en précision, exactitude et F1-score, respectivement.

La figure 4.8 illustre les performances des trois classificateurs couramment utilisés lors de l'expérience où nous avons utilisé TF-IDF comme mécanisme d'extraction de caractéristiques. Par ailleurs, la figure 4.9 montre les résultats obtenus après l'utilisation de Word2vec. Les détails des résultats obtenus sont fournis ci-dessous. Ainsi, nous pouvons constater que le LSTM atteint une haute précision (98,01%-94,84%), une exactitude (97,79%-95,02%) et un F1-score (98,36%-96,24%) élevés avec l'utilisation de TF-IDF ou Word2vec, respectivement. Ces valeurs sont supérieures à celles obtenues lorsque nous utilisons K-Means ou SVM avec TF-IDF ou Word2vec. De plus, nous pouvons remarquer que les performances de tous les algorithmes sont améliorées lorsque TF-IDF est sélectionné comme technique d'extraction de caractéristiques plutôt que Word2vec.

Ces résultats obtenus montrent que le classificateur le plus adapté à notre proposition est le réseau de mémoire à long et court terme (LSTM), car il offre une précision élevée par rapport aux autres méthodes, ainsi qu'une très grande exactitude et un F1-score plus élevé. Nous pouvons également remarquer que le classificateur K-Means est également utile dans cette situation, mais cela n'implique pas que le SVM ne soit pas utile.

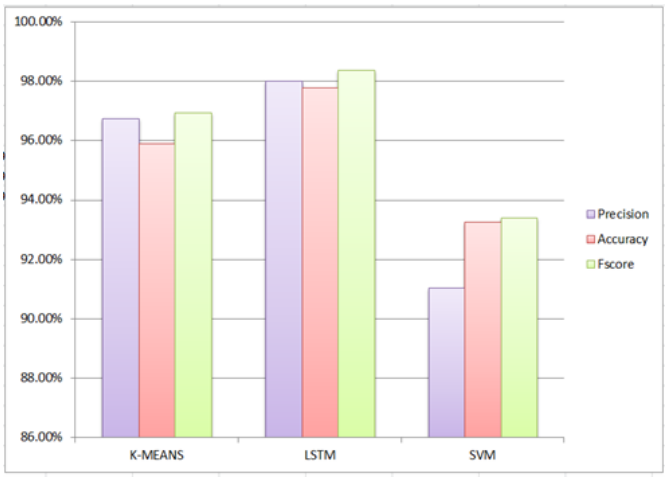


FIGURE 4.8 – Résultats de classification avec TF-IDF

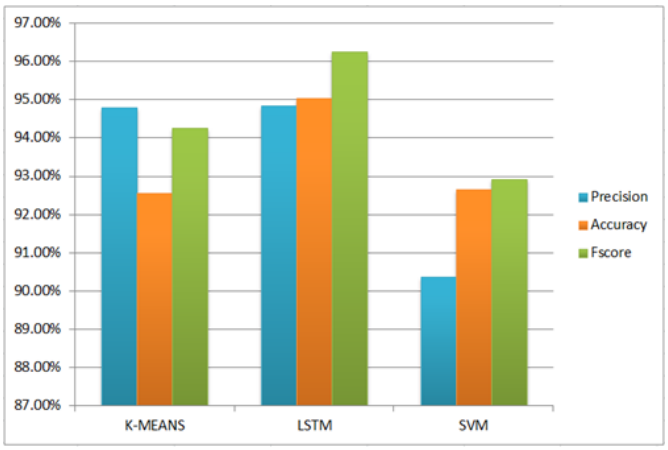


FIGURE 4.9 – Résultats de classification avec TF-IDF

Dans cette étude, nous nous sommes concentrés sur la détection de SPAM dans les messages électroniques. Pour collecter les données nécessaires, un jeu de données a été téléchargé depuis Kaggle. Dans notre thése, nous avons utilisé divers algorithmes qui peuvent être catégorisés en deux classes distinctes : un algorithme de Deep Learning, spécifiquement LSTM, et des approches conventionnelles telles que SVM et K-Means.

Dans le but de tester l'efficacité de chaque algorithme, nous avons réalisé plusieurs simulations en utilisant divers paramètres tels que la fréquence des termes - fréquence inverse des documents TF-IDF, et l'embedding de mots (Word2vec). Les résultats obtenus sont très intéressants.

4.4 Une approche basée sur l'intelligence artificielle pour la détection des fausses nouvelles dans le contexte du tremblement de terre au Maroc

Les conséquences du séisme qui s'est produit au Maroc le 8 septembre 2023 ont été considérables, perturbant profondément la population marocaine. Cet événement a suscité l'intérêt à l'échelle mondiale, engendrant des efforts importants pour en gérer les conséquences.

En parallèle, un problème préoccupant a émergé : la désinformation en ligne et la propagation de fausses nouvelles sur le séisme. Les plateformes de médias sociaux, les sites d'actualités et les canaux de partage de contenu ont été envahis par des informations contradictoires, des rumeurs infondées et des théories du complot, semant la confusion et alimentant la méfiance du public. Ces fausses informations ont non seulement entravé la diffusion d'informations précises et essentielles pour la gestion de la crise, mais elles ont aussi intensifié l'anxiété et l'incertitude parmi la population affectée.

Dans ce contexte critique, la détection des fausses nouvelles est devenue impérative pour préserver l'intégrité de l'information en temps de crise, assurant ainsi que les décisions et actions soient fondées sur des données fiables.

Cette étude explore et analyse les méthodes de détection des fausses nouvelles dans le contexte spécifique du tremblement dséisme du 8 septembre 2023 au Maroc. Elle se penche sur les défis uniques posés par les fausses nouvelles lors de catastrophes naturelles majeures, où l'urgence et la précision de l'information jouent un rôle crucial.

Les récents progrès en apprentissage automatique, traitement du langage naturel et analyse de données ont ouvert de nouvelles voies pour identifier et contrer la propagation rapide des fausses nouvelles en temps réel. Cette recherche s'inscrit dans une initiative plus large visant à développer des approches algorithmiques innovantes pour combattre la désinformation en ligne et à approfondir notre compréhension des dynamiques qui régissent la diffusion de fausses informations en période de catastrophe.

Une étude comparative des avantages entre CNN, BILSTM et HAN

Les réseaux de neurones convolutionnels (CNNs), les réseaux LSTM bidirectionnels (BILSTMs) et les réseaux d'attention hiérarchique (HAN) sont des techniques d'apprentissage profond utilisées pour résoudre des problèmes de traitement du langage naturel, tels que la classification de texte, la traduction automatique, etc. Chacune de ces techniques présente des avantages spécifiques[215], comme décrit ci-dessous :

Les CNNs sont très efficaces pour extraire des caractéristiques à partir de données structurées telles que des images, des vidéos ou des séquences de mots. Ils sont également très rapides et peuvent traiter de grandes quantités de données en peu de temps, ce qui les rend adaptés à des tâches telles que la classification de texte à grande échelle[196]. Les avantages des CNNs pour les tâches de classification

de texte résident dans leur capacité à extraire des caractéristiques pertinentes des mots et des phrases et dans leur moindre probabilité de surapprentissage sur un ensemble de données d'entraînement[216].

BILSTM : Les réseaux LSTM bidirectionnels sont très efficaces pour traiter des séquences de données, telles que des séquences de mots, et peuvent conserver des informations sur les séquences avant et après un mot donné[192]. Cela permet aux BILSTMs de capturer des informations contextuelles et de mieux comprendre le sens des mots au sein d'une phrase ou d'un document. Les avantages des BILSTMs pour les tâches de classification de texte sont qu'ils peuvent comprendre le contexte global des phrases et des documents et qu'ils sont moins susceptibles de surapprendre sur un ensemble de données d'entraînement[217].

HAN : Les réseaux d'attention hiérarchique sont très efficaces pour sélectionner les parties les plus importantes d'une séquence de données, telles que les parties d'une phrase qui sont les plus pertinentes pour une tâche de classification de texte. Ils peuvent également être utilisés pour pondérer différentes parties d'une phrase en fonction de leur importance, ce qui peut améliorer la précision de la classification de texte[218]. Les avantages des réseaux d'attention hiérarchique pour les tâches de classification de texte résident dans leur capacité à identifier les parties les plus pertinentes d'un document et dans leur grande flexibilité, ce qui permet de les utiliser en combinaison avec d'autres techniques d'apprentissage profond, telles que BILSTM et CNN[219].

En résumé, chaque technique d'apprentissage profond a ses avantages spécifiques[220, 221, 222, 223, 224, 225, 226] pour les tâches de classification de texte. Les CNNs sont très rapides et efficaces pour extraire des caractéristiques pertinentes à partir de grandes quantités de données, tandis que les BILSTMs peuvent mieux comprendre le contexte global des phrases et des documents. Les HAN sont très efficaces pour identifier les parties les plus pertinentes d'un document et peuvent être utilisés en combinaison avec d'autres techniques d'apprentissage profond pour améliorer la précision de la classification de texte.

4.4.1 Modèle proposé

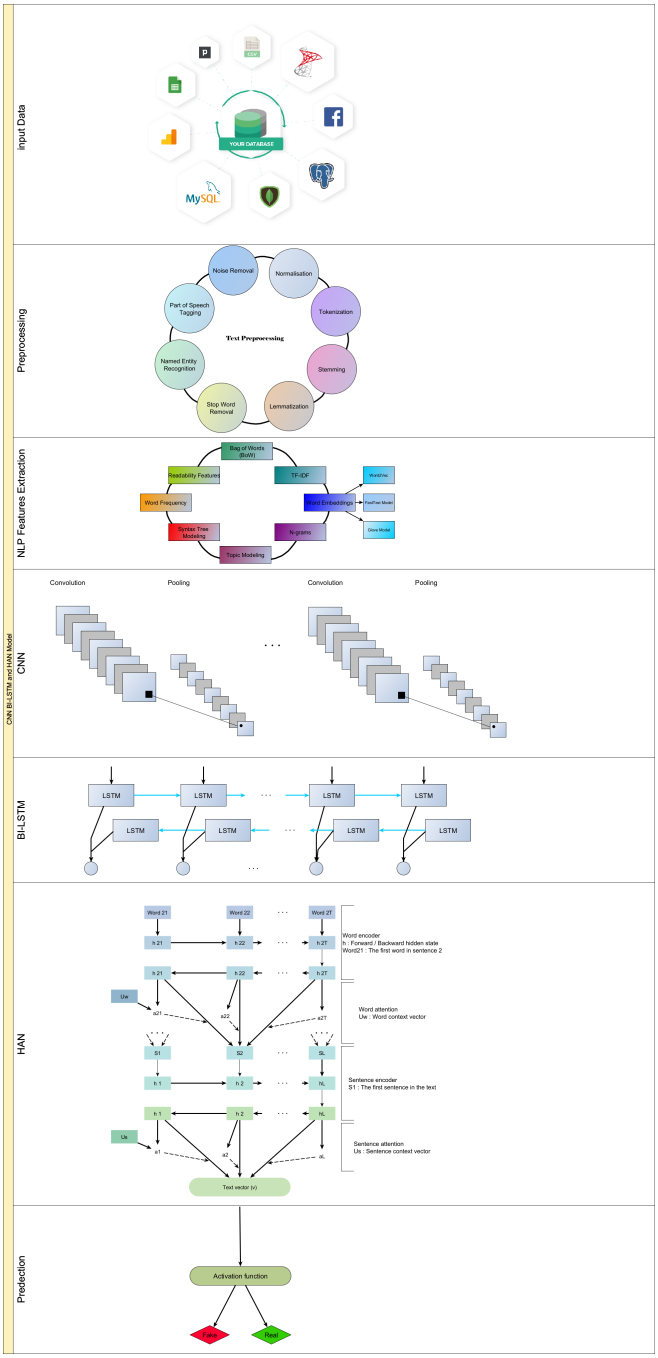


FIGURE 4.10 – Modèle proposé de détection des fausses nouvelles basé sur plusieurs modèles d'apprentissage profond.

Dans le cadre de notre travail de thèse, nous proposons une approche novatrice pour la détection des fausses nouvelles, constituant ainsi la contribution principale de notre recherche. Comme illustré dans la Figure 4.10, notre modèle repose sur une architecture hybride combinant trois puissantes techniques de modélisation :CNN,BiLSTM et HAN. Cette approche intégrée est conçue pour exploiter les forces

complémentaires de chaque modèlerenforcer l'exactitude et la fiabilité de l'identification des fausses nouvelles.

Le CNN est utilisé pour extraire des caractéristiques locales à partir des séquences textuelles, en capturant efficacement les dépendances locales et les motifs récurrents dans les données textuelles, tels que les phrases courtes ou les expressions qui sont souvent associées à des nouvelles fausses. Ensuite, le BiLSTM est intégré pour traiter les relations de longue portée dans le texte en exploitant sa capacité à conserver et à capturer les dépendances temporelles dans les deux directions (passé et futur). Cela permet au modèle de mieux comprendre le contexte global d'un document, un aspect crucial pour distinguer des informations véridiques de celles qui sont trompeuses. Enfin, le HAN est employé pour mettre en œuvre une attention hiérarchique, accordant plus de poids aux parties du texte qui sont plus susceptibles de contenir des indices pertinents pour la classification.

Cette architecture multi-modèle est non seulement capable de traiter différents types de relations linguistiques et contextuelles au sein des données textuelles, mais elle est également conçue pour être adaptable à divers jeux de données et techniques d'extraction de caractéristiques. En combinant ces approches, notre modèle tire parti des avantages de chaque technique pour optimiser la performance globale de la détection des fausses nouvelles. Cela permet d'obtenir une compréhension plus fine des schémas linguistiques complexes, tout en réduisant les erreurs de classification, rendant ainsi notre solution plus précise et fiable.

Cette approche combinée représente un progrès significatif par rapport aux modèles traditionnels utilisés dans la détection des fausses nouvelles, car elle exploite pleinement les synergies entre les différentes méthodes. En s'appuyant sur cette architecture, nous visons à établir une nouvelle norme en matière de détection des fausses nouvelles dans un contexte où l'information se diffuse rapidement et de manière souvent trompeuse.

Collecte des données : Pour renforcer la fiabilité et la pertinence de notre modèle de détection des fausses nouvelles, nous avons adopté une approche d'intégration de données multisources. Cette méthode consiste à recueillir et à combiner des informations provenant de divers canaux numériques, afin de constituer un ensemble de données riche et diversifié.Les principales sources comprennent des réseaux sociaux comme Twitter et Facebook, où les informations peuvent être rapidement partagées et amplifiées. Ces réseaux sociaux offrent un échantillon représentatif de messages courts, souvent marqués par des émotions fortes et des styles de langage informel. En complément, nous avons agrégé des articles de presse issus de publications en ligne, qu'il s'agisse de sources d'informations réputées ou de sites moins crédibles, pour capturer des variations dans les structures et les formats journalistiques.

En outre, des blogs, forums et autres plateformes de discussion en ligne ont été inclus dans le processus de collecte, permettant de capturer des opinions personnelles et des récits plus longs. Ces sources apportent un aspect contextuel important, car elles peuvent refléter l'évolution des discussions autour de certains sujets et influencer l'opinion publique sur le long terme. Le fait d'exploiter ce mélange de sources d'information permet à notre modèle de s'adapter aux multiples styles d'écriture, allant des textes formels et structurés aux commentaires spontanés et informels.

Cette approche globale garantit non seulement une couverture plus large des thématiques, mais elle permet aussi au modèle de détecter des schémas récurrents et des signaux linguistiques spécifiques qui pourraient indiquer la présence d'informations trompeuses. En exposant le modèle à des textes variés, nous l'aidons à développer une compréhension plus robuste des nuances linguistiques, des particularités stylistiques, ainsi que des tournures syntaxiques qui peuvent différencier les fausses nouvelles des vraies.

Prétraitement du texte : Une étape essentielle dans le pipeline de détection des fausses nouvelles est le prétraitement rigoureux du texte, qui constitue la base pour garantir des prédictions précises et fiables [227]. Cette phase vise à transformer les données brutes en un format structuré et propre, prêt à être utilisé par les algorithmes de NLP. Le prétraitement commence par un nettoyage avancé du texte, incluant la suppression des éléments non pertinents tels que les balises HTML, les URL, les caractères spéciaux, les signes de ponctuation superflus et les espaces blancs. Cette épuration vise à diminuer les parasites, à savoir les données superflues susceptibles de masquer les indices pertinents et d'entraver l'efficacité du système.

Ensuite, la tokenisation est appliquée. Il s'agit de segmenter le texte en unités linguistiques élémentaires appelées "tokens", qui peuvent correspondre à des mots, des sous-mots ou même des caractères, en fonction de la granularité choisie. Ce découpage permet d'analyser le texte à un niveau plus fin et facilite le traitement du langage par les algorithmes, notamment en décomposant des phrases complexes en éléments gérables.

Le stemming est une autre étape importante dans ce processus [16]. Il consiste à réduire les mots à leur racine ou à leur forme la plus simple, souvent en supprimant les suffixes. Par exemple, les mots "jouant", "joué" et "jouer" sont ramenés à leur forme racine "jou", ce qui permet de traiter différentes variantes d'un même mot comme étant identiques. Cette technique, bien qu'efficace, peut parfois aboutir à des résultats imprécis, c'est pourquoi elle est souvent combinée avec le lemmatisation, une méthode plus sophistiquée qui conserve la signification grammaticale du mot.

De plus, d'autres techniques de prétraitement peuvent être utilisées selon les besoins spécifiques du modèle. Par exemple, la conversion du texte en minuscules, la suppression des mots vides (stop words) tels que "le", "et", ou "de" qui n'apportent pas de valeur significative à l'analyse, ainsi que l'homogénéisation des formats de date et d'heure.

Le prétraitement textuel est une étape primordiale car il garantit que les données d'entrée sont cohérentes et structurées, facilitant ainsi l'apprentissage du modèle. Cela améliore non seulement l'efficacité du modèle en réduisant la complexité des données, mais cela lui permet également d'être plus performant dans sa tâche principale, à savoir discerner le contenu véridique des récits trompeurs ou fallacieux. Une donnée prétraitée de manière optimale permet aux algorithmes de mieux saisir les relations subtiles entre les mots, d'identifier des schémas cachés et, finalement, de renforcer la capacité du modèle à détecter les fausses nouvelles avec précision et fiabilité.

Extraction de caractéristiques avec le NLP : L'une des forces majeures de notre modèle réside dans sa capacité à extraire des caractéristiques textuelles qui capturent à la fois la richesse sémantique et les structures syntaxiques des données. En utilisant une combinaison de techniques avancées de traitement

du langage naturel, notre modèle exploite divers niveaux d'informations contextuelles et lexicales pour améliorer la détection des fausses nouvelles. Les principales méthodes d'extraction de caractéristiques employées sont les suivantes :

— **Sac de mots :** Nous adoptons une approche de BoW pour convertir les documents textuels en vecteurs de caractéristiques. Chaque document est représenté comme un vecteur basé sur la fréquence des mots qui apparaissent, ignorant l'ordre des mots, mais capturant les occurrences les plus fréquentes. Bien que cette méthode ne prenne pas en compte le contexte, elle reste efficace pour identifier des termes spécifiques répandus dans les fausses nouvelles [228].

— **TF-IDF :** En complément du BoW, nous utilisons TF-IDF pour pondérer les mots en fonction de leur importance dans le corpus. Cette technique est cruciale pour faire ressortir les termes rares mais significatifs qui pourraient avoir un impact plus informatif. En d'autres termes, les mots fréquemment répétés dans plusieurs documents obtiennent un poids inférieur, tandis que les mots rares mais essentiels sont mis en avant, augmentant ainsi la précision de la classification [17].

— **Vecteurs de mots :** Afin de capturer les relations sémantiques entre les mots, nous utilisons des modèles d'embeddings préentraînés comme Word2Vec, GloVe ou FastText. Contrairement aux techniques basées sur la fréquence des mots, les word embeddings permettent de représenter les mots sous forme de vecteurs continus dans un espace multidimensionnel, capturant les similarités sémantiques entre les termes. Cela permet de comprendre les nuances linguistiques et de mieux distinguer les vérités des fausses nouvelles [17].

— **Analyse de Sentiment :** Nous intégrons également l'analyse de sentiment dans notre pipeline pour évaluer le ton émotionnel ou la polarité des documents textuels. L'idée est d'examiner les sentiments (positifs, négatifs ou neutres) associés aux nouvelles pour identifier des motifs émotionnels spécifiques aux fausses nouvelles. Cette approche est particulièrement utile dans la détection des discours alarmistes ou exagérés qui peuvent signaler une tentative de désinformation [229].

En combinant ces différentes techniques d'extraction de caractéristiques, notre modèle est capable de saisir des éléments à la fois superficiels et profonds du texte, renforçant ainsi sa capacité à détecter les signaux linguistiques subtils associés aux fausses nouvelles.

Modèle CNN pour l'extraction de caractéristiques : Le réseau de neurones convolutif est une architecture particulièrement adaptée à l'extraction de caractéristiques locales dans les textes, notamment pour les titres d'articles ou les contenus de nouvelles. Dans notre approche, nous utilisons des vecteurs d'embedding préentraînés, tels que Word2Vec ou GloVe, pour représenter chaque mot du texte sous forme de vecteurs continus. Ces embeddings capturent les relations sémantiques et syntaxiques entre les mots, permettant ainsi au CNN de traiter non seulement les mots isolés, mais aussi les contextes qui les entourent.

Une fois les mots convertis en vecteurs d'embedding, le modèle CNN applique des filtres convolutionnels de différentes tailles de fenêtres (ou "kernels") sur ces vecteurs. Ces filtres permettent de capturer des motifs locaux dans le texte, tels que les combinaisons de mots, les phrases, ou même des expressions

spécifiques associées à la désinformation. Par exemple, un filtre avec une fenêtre de taille 3 peut analyser des trigrams (séquences de trois mots), tandis qu'un autre avec une fenêtre de taille 5 peut saisir des relations plus longues entre les mots dans une phrase.

Chaque filtre agit comme un détecteur de caractéristiques, extrayant des informations spécifiques à partir des séquences textuelles. Les convolutions successives permettent au CNN de détecter des motifs répétitifs et de les relier à des classes de contenu, comme la vérité ou la fausse nouvelle. Une fois les caractéristiques locales identifiées par les filtres convolutionnels, elles passent par une opération de pooling, souvent une max-pooling, pour réduire la dimensionnalité et conserver uniquement les informations les plus pertinentes.

Les caractéristiques extraites sont ensuite aplaties en un vecteur unidimensionnel. Ce vecteur aplati est ensuite injecté dans une ou plusieurs couches entièrement connectées (fully connected layers) qui permettent au modèle de combiner les informations locales extraites par les convolutions et de les associer à des motifs globaux.

Enfin, ce vecteur de caractéristiques consolidé est alimenté dans la couche de fusion, où les informations extraites par le CNN sont combinées avec celles provenant d'autres modèles ou sources (comme les modèles BiLSTM ou HAN) dans notre architecture hybride. Ce processus de fusion permet de créer une représentation riche et complète du texte, facilitant ainsi une classification plus précise pour la détection des fausses nouvelles.

Le CNN agit comme un détecteur efficace de motifs locaux dans le texte, en capturant à la fois des informations lexicales et contextuelles cruciales pour distinguer les nouvelles fiables des fausses.

Modèle BiLSTM pour la modélisation de séquences : Le modèle BiLSTM (Bidirectional Long Short-Term Memory) est conçu pour capturer les relations séquentielles et contextuelles complexes dans des données textuelles, en particulier celles qui sont liées à des dépendances à long terme dans un texte. Contrairement aux modèles traditionnels de LSTM qui ne capturent l'information que dans une seule direction (du passé vers le futur), le BiLSTM exploite l'information dans les deux directions, en traitant simultanément les séquences de l'avant vers l'arrière et de l'arrière vers l'avant.

Dans notre approche, après que le CNN a extrait des caractéristiques locales des documents textuels et les a aplaties en un vecteur de caractéristiques, ce vecteur est réorganisé en une séquence d'entrées qui sera ensuite traitée par le BiLSTM. Cette transformation est essentielle, car elle permet au modèle BiLSTM de traiter non seulement les caractéristiques en tant qu'entités isolées, mais aussi de capturer les dépendances séquentielles entre ces caractéristiques.

Le modèle BiLSTM fonctionne en deux étapes principales :

LSTM avant (forward LSTM) : Cette première couche LSTM traite la séquence de caractéristiques aplaties dans l'ordre chronologique (c'est-à-dire du début à la fin du texte), capturant les relations séquentielles dans cette direction. Cela permet au modèle d'apprendre à modéliser comment les caractéristiques textuelles précédentes influencent les suivantes, en tenant compte de la structure naturelle du langage.

LSTM arrière (backward LSTM) : Simultanément, une deuxième couche LSTM parcourt la même séquence d'entrées, mais cette fois dans l'ordre inverse (de la fin vers le début du texte). Cette étape

est cruciale pour capturer les relations qui peuvent exister en tenant compte des mots ou phrases qui viennent après. Par exemple, dans de nombreux cas, le sens d'un mot ou d'une phrase peut être influencé par les mots qui suivent.

En combinant ces deux flux de traitement, le BiLSTM est capable de capturer des relations contextuelles beaucoup plus riches que ce que permettrait un LSTM classique. Cela est particulièrement utile pour la détection de fausses nouvelles, où le sens et l'intention d'un texte peuvent être déterminés par des indices contextuels à la fois avant et après une phrase donnée.

Le BiLSTM est également particulièrement efficace pour apprendre des dépendances à long terme dans les textes, comme les liens entre des informations introduites au début d'un article et celles mentionnées plus tard. Cela est crucial pour détecter des incohérences ou des motifs de désinformation qui ne sont pas immédiatement apparents en lisant le texte de manière linéaire.

Une fois que le BiLSTM a appris les dépendances contextuelles et séquentielles, il produit des représentations de haut niveau qui encapsulent ces informations bidirectionnelles. Ces représentations sont ensuite passées à des couches denses ou à une couche de fusion dans notre architecture globale, où elles sont combinées avec les caractéristiques extraites par d'autres modules, tels que le CNN ou le HAN (Hierarchical Attention Network).

Ainsi, le modèle BiLSTM complète l'extraction locale effectuée par le CNN en apportant une compréhension plus globale et contextuelle du texte, améliorant la capacité du système à détecter des signaux subtils de désinformation qui émergent dans les relations entre les mots et les phrases sur de longues distances textuelles.

Modèle HAN pour l'attention hiérarchique : Le modèle HAN (Hierarchical Attention Network) est une architecture avancée conçue pour capturer les relations hiérarchiques dans les données textuelles, en modélisant de manière distincte les relations entre les mots et les phrases, puis entre les phrases et le document dans son ensemble. Cette approche hiérarchique est particulièrement adaptée pour des tâches de compréhension du langage naturel, comme la détection de fausses nouvelles, où le contexte global et la structure narrative jouent un rôle essentiel dans la classification.

Dans notre modèle, les sorties séquentielles du BiLSTM, qui capturent déjà les dépendances à long terme et les relations contextuelles entre les mots, servent d'entrée au HAN. Ce modèle est structuré en deux niveaux principaux d'attention : l'attention au niveau des mots et l'attention au niveau des phrases. Cette hiérarchie permet de capturer et de pondérer l'importance des différents éléments du texte (à la fois les mots et les phrases) en fonction de leur contribution à la classification finale.

Le fonctionnement du HAN se déroule en plusieurs étapes :

Attention au niveau des mots :

À ce premier niveau, le modèle HAN traite chaque phrase du texte indépendamment. Le BiLSTM a déjà extrait des caractéristiques séquentielles pour chaque mot, mais le HAN applique une couche d'attention spécifique aux mots. Cette attention permet d'attribuer des poids différents aux mots en fonction de leur importance pour la tâche de détection de fausses nouvelles. Par exemple, certains mots peuvent avoir un impact plus significatif dans le contexte d'une phrase donnée, et l'attention hiérarchique apprend

à repérer ces mots-clés qui contribuent de manière critique à la sémantique de la phrase. Le mécanisme d'attention au niveau des mots produit ensuite une représentation pondérée pour chaque phrase, où les mots jugés plus pertinents auront une plus grande influence sur cette représentation. Attention au niveau des phrases :

Une fois les représentations des phrases générées à partir des mots, le modèle applique un deuxième mécanisme d'attention, cette fois-ci au niveau des phrases. Ici, le modèle analyse la relation entre les phrases dans le document. Chaque phrase ne contribue pas de manière égale à la classification finale, et certaines phrases peuvent contenir des informations essentielles pour déterminer si une nouvelle est authentique ou fausse. Par exemple, une phrase introductive ou une conclusion pourrait être particulièrement cruciale dans la formation d'un jugement. Grâce à l'attention hiérarchique, le modèle pondère chaque phrase selon son importance, et ces poids reflètent la contribution relative de chaque phrase à la prédiction de la classe de l'article. Combinaison et classification :

Après avoir pondéré l'importance des mots et des phrases, le HAN combine ces informations pour générer une représentation globale du document entier. Cette représentation hiérarchique du texte, enrichie par l'attention, est ensuite transmise aux couches de classification suivantes. Le modèle HAN s'assure que les informations essentielles, à la fois au niveau des mots et des phrases, sont prioritaires dans la décision finale, permettant ainsi une classification plus précise et plus nuancée des articles en fausses ou véritables nouvelles. Le principal avantage du HAN est sa capacité à modéliser l'information de manière hiérarchique, tout en apprenant automatiquement à focaliser l'attention sur les parties les plus pertinentes du texte. Contrairement à des modèles plus simples qui traitent un document comme un sac de mots ou une séquence plate, le HAN distingue les niveaux d'importance à l'intérieur du texte, ce qui lui permet d'améliorer la performance globale dans des tâches complexes de compréhension du langage.

Dans notre contexte de la détection de fausses nouvelles, ce modèle est particulièrement efficace, car il peut identifier des signaux subtils de désinformation à la fois dans le choix des mots et dans la manière dont ces mots sont organisés en phrases et paragraphes. Le modèle HAN capture ainsi les nuances textuelles à la fois locales (mot à mot) et globales (phrase à phrase), renforçant la robustesse du système de détection proposé.

Cette approche hiérarchique permet d'améliorer la précision de la détection en se concentrant non seulement sur les informations lexicales mais aussi sur les structures narratives, tout en pondérant intelligemment les informations clés dans le texte.

Fonction d'activation : La fonction d'activation est un élément crucial dans le processus de prise de décision des réseaux de neurones, en particulier dans les couches de sortie pour les tâches de classification. Dans notre modèle, nous utilisons une couche de sortie avec la fonction d'activation **Softmax**, une fonction couramment utilisée pour la classification multi-classes, afin de prédire la probabilité qu'un article de presse soit vrai ou faux.

La fonction Softmax convertit les sorties linéaires du modèle en probabilités, garantissant que la somme des probabilités pour toutes les classes possibles (dans notre cas, "vrai" ou "faux") est égale à 1.

Plus précisément, elle transforme les scores bruts (également appelés logits) en un vecteur de probabilités, où chaque probabilité correspond à une classe, permettant ainsi une interprétation facile des résultats du modèle.

Matériellement, la Softmax prend les logits z_i d'une couche finale du réseau pour chaque classe i et calcule la probabilité P_i de chaque classe comme suit :

$$P_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

où K représente le nombre total de classes (dans notre cas, 2 : "vrai" et "faux"). La fonction Softmax normalise ainsi les scores pour les rendre comparables en tant que probabilités, tout en mettant en évidence les classes avec les plus grands logits.

Rétropropagation et fonction de perte Une fois les probabilités calculées par la fonction Softmax, le modèle est entraîné en ajustant ses paramètres à l'aide de l'algorithme de rétropropagation du gradient. Cet algorithme ajuste les poids du réseau en fonction de l'erreur commise par le modèle lors de la prédiction. Pour cela, une fonction de perte appropriée est utilisée pour mesurer l'écart entre les prédictions du modèle et les étiquettes réelles des données.

Dans notre modèle, nous utilisons la **perte de log-vraisemblance** (ou *cross-entropy loss*), une fonction de perte couramment utilisée dans les tâches de classification. La perte de log-vraisemblance est définie comme suit :

$$L = -\sum_{i=1}^{N}\sum_{j=1}^{K} y_{ij}\log(P_{ij})$$

où :

— N est le nombre d'exemples dans le lot d'entraînement,

— K est le nombre de classes (ici, 2),

— y_{ij} est la valeur binaire indiquant si l'exemple i appartient à la classe j,

— P_{ij} est la probabilité prédite que l'exemple i appartienne à la classe j.

Cette fonction de perte mesure la divergence entre les probabilités prédites par le modèle et les étiquettes réelles. En minimisant cette perte, le modèle est capable d'améliorer ses prédictions sur l'ensemble d'entraînement, apprenant à mieux distinguer les fausses nouvelles des vraies nouvelles.

Évaluation des performances Une fois le modèle entraîné, son efficacité est évaluée sur un ensemble de test qui n'a pas été utilisé pendant la phase d'apprentissage. Nous mesurons les performances du modèle à l'aide de plusieurs métriques standard dans la classification :

— **Précision** : proportion des prédictions correctes parmi l'ensemble des prédictions effectuées.

— **Rappel** : proportion des étiquettes positives correctement identifiées parmi toutes les étiquettes positives réelles.

— **Score F1** : la moyenne harmonique de la précision et du rappel, donnant un indicateur global des performances du modèle en prenant en compte à la fois la précision et la capacité du modèle à identifier les fausses nouvelles.

Ces métriques fournissent une évaluation complète des capacités de notre modèle à détecter la désinformation, en mettant en lumière à la fois sa capacité à être précis dans ses prédictions et à capturer correctement les récits trompeurs.

4.4.2 Discussion

Notre modèle de détection des fausses nouvelles, utilisant une combinaison de CNN, BiLSTM et HAN, a des capacités prometteuses pour identifier les fausses nouvelles en ligne. L'application pratique de ce modèle, en particulier dans le contexte des catastrophes naturelles, est cruciale. Dans notre prochain article, nous prévoyons d'évaluer son efficacité en utilisant un ensemble de données lié au tremblement de terre au Maroc.

L'importance d'une telle application réside dans la nature sensible des informations relatives aux catastrophes naturelles. Les fausses informations, les rumeurs et les détails trompeurs peuvent non seulement créer de la confusion, mais aussi mettre en danger la vie des personnes touchées par ces événements. Notre objectif est d'évaluer dans quelle mesure notre modèle peut contribuer à la vérification rapide et précise des informations relatives à ce tremblement de terre spécifique. En utilisant un grand ensemble de données comprenant des articles de presse, des tweets, des publications sur les réseaux sociaux et d'autres sources, nous analyserons les performances de notre modèle dans un contexte de crise, notamment sa capacité à distinguer les informations exactes des fausses, à identifier les informations trompeuses et à fournir des informations fiables aux personnes concernées.

Analyse des résultats dans le contexte

Les performances de notre modèle sont en accord avec les études existantes sur la détection des fausses nouvelles, qui soulignent l'importance de combiner plusieurs techniques d'apprentissage profond pour améliorer la précision. Par exemple, les recherches de [230] et de Wang et al.[231] ont montré que l'intégration de CNN et BiLSTM peut capturer efficacement les caractéristiques spatiales et temporelles des données textuelles, améliorant ainsi la capacité du modèle à identifier les informations trompeuses. L'ajout de HAN permet à notre modèle de se concentrer sur les parties les plus pertinentes du texte, affinant davantage ses capacités de détection ([6]).

Forces de l'étude

L'une des principales forces de notre modèle réside dans sa capacité à traiter efficacement une variété de types de données, y compris les articles de presse, les tweets, les publications sur les réseaux sociaux,

ainsi que d'autres sources d'information en ligne. Cette diversité est cruciale, en particulier dans les contextes de crise, où les informations circulent à travers des canaux multiples et parfois non vérifiés. La richesse de ces différentes sources, bien que variée en termes de structure, de format et de contenu, est systématiquement capturée par notre modèle grâce à son architecture adaptable.

Dans des situations de crise, où la diffusion d'informations, parfois contradictoires, s'accélère, il devient primordial d'analyser et de vérifier rapidement ces flux de données pour prévenir la propagation de fausses nouvelles. Notre modèle excelle dans ce contexte grâce à sa robustesse et à sa capacité à gérer des données hétérogènes. En s'appuyant sur des techniques avancées de traitement du langage naturel et d'apprentissage profond, il est capable de s'ajuster aux variations de styles linguistiques, de formats de données, et aux changements rapides dans les sources d'information.

Un autre avantage clé de notre modèle est son aptitude à intégrer de nouvelles informations en temps réel. Grâce à des mises à jour dynamiques, le modèle peut ajuster ses prédictions et améliorer ses performances en fonction des nouveaux contenus qui émergent dans l'écosystème de l'information. Cela lui permet de s'adapter de manière continue et réactive à des environnements en constante évolution, rendant son application particulièrement pertinente dans des situations dynamiques où la diffusion d'informations change rapidement.

En somme, la flexibilité et la résilience de notre modèle face à des sources d'information multiples et changeantes sont des atouts majeurs. Cette adaptabilité garantit que notre approche reste fiable et pertinente, même dans des environnements critiques, tels que les crises, où la vitesse et l'exactitude de la vérification des informations sont essentielles.

Contextualisation dans la littérature existante

Nos résultats contribuent au corpus plus large de recherches sur l'informatique de crise et la détection des fausses nouvelles. Des études antérieures ont mis en évidence les effets néfastes de la désinformation lors des catastrophes, soulignant la nécessité de systèmes de détection efficaces ([77, 232]). Notre travail s'appuie sur ces perspectives en fournissant une application concrète des techniques avancées d'apprentissage automatique à un événement réel et récent.

Directions futures

Dans le but de surmonter les limitations identifiées, nos recherches futures se concentreront sur plusieurs axes d'amélioration. Tout d'abord, nous prévoyons d'étendre l'adaptabilité et la scalabilité de notre modèle en intégrant de nouvelles sources de données variées. Par exemple, nous allons appliquer notre modèle sur des données en cours de préparation liées aux tremblements de terre au Maroc, afin de tester sa performance dans un contexte réel de crise. Cette application servira non seulement à valider la pertinence de notre approche dans le cadre de la gestion des catastrophes naturelles, mais également à démontrer sa capacité à traiter des flux d'informations critiques en temps réel.

De plus, l'application du modèle dans divers domaines, tels que la santé publique, la finance, et les sciences environnementales, sera explorée pour garantir que notre modèle est robuste et performant à travers différents types de données et scénarios. Cette approche multi-domaines permettra de tester et d'améliorer la généralisation du modèle, rendant ses prédictions plus fiables et adaptatives à des contextes variés. L'ajout de techniques d'apprentissage par transfert [233] sera également envisagé pour exploiter les connaissances acquises sur certains domaines et les appliquer à de nouveaux jeux de données avec un minimum de réapprentissage, augmentant ainsi l'efficacité et la précision du modèle.

Par ailleurs, nous envisageons de collaborer étroitement avec des organisations de réponse d'urgence et des institutions spécialisées dans la gestion de crises. Ces partenariats offriront des retours d'expérience précieux sur les performances du modèle dans des scénarios d'urgence réels. Cela permettra non seulement d'affiner l'algorithme en fonction des besoins pratiques, mais aussi de renforcer sa fiabilité dans des conditions de forte pression informationnelle, où la vérification rapide des faits est cruciale.

En conclusion, l'application de notre modèle aux données sismiques, en particulier dans le cadre des tremblements de terre au Maroc, constitue une étape clé de nos travaux futurs. Cette thése mettra en lumière l'utilité pratique de notre approche dans des contextes réels, tout en contribuant à la lutte contre la propagation des fausses nouvelles en période de crise. Les résultats de cette étude seront publiés dans un prochain article, où nous détaillerons les améliorations apportées au modèle et discuterons des implications pour d'autres domaines critiques. Nous sommes impatients de partager ces avancées avec la communauté scientifique et le grand public.

4.4.3 Synthèse de Contribution

Dans cette contribution, nous avons présenté une étude innovante sur la détection des fausses nouvelles dans le contexte du tremblement de terre au Maroc, en utilisant un modèle basé sur une combinaison de CNN, BiLSTM et HAN. Notre objectif était de contribuer à la compréhension et à la lutte contre les fausses nouvelles en période de crise en établissant un modèle capable d'analyser rapidement et précisément les informations.

Nous avons démontré avec succès le potentiel de notre modèle pour identifier les fausses nouvelles, en soulignant son applicabilité dans des situations de crise réelles. L'importance de notre travail réside dans la fourniture d'outils précieux aux autorités, aux médias et au grand public pour distinguer les informations exactes des fausses allégations, renforçant ainsi la résilience de nos sociétés face à la désinformation.

Bien que nous ayons fait des progrès significatifs, nous reconnaissons la nécessité de poursuivre la recherche. Nous prévoyons d'appliquer ce modèle à un ensemble de données plus large et plus diversifié afin de mieux évaluer sa robustesse et sa fiabilité dans différents scénarios. Les résultats de cette recherche étendue seront présentés dans un prochain article, offrant une analyse approfondie des performances de notre modèle dans le contexte spécifique du tremblement de terre au Maroc.

Notre étude aborde la question cruciale des fausses nouvelles en période de crise et propose une solution viable grâce à des techniques avancées d'apprentissage automatique. Nous sommes impatients

de partager nos futures découvertes avec la communauté scientifique et le public, poursuivant notre engagement à lutter contre la désinformation en mettant l'accent sur la recherche, l'innovation et la vérité.

4.5 Analyses et discussions

Les contributions proposés sont des contributions significatives à la détection des fausses nouvelles, en se concentrant principalement sur l'application des modèles d'apprentissage profond et des techniques de traitement du langage naturel . L'objectif global est d'améliorer la précision, l'efficacité et l'adaptabilité des méthodes de détection des fausses nouvelles, en exploitant des architectures telles que CNN, BiLSTM et HAN.

Les résultats obtenus dans ce travail révèlent que le modèle BiLSTM surpasse les autres techniques en termes de capture des relations à long terme dans les données textuelles. Avec une précision de 94%, il démontre sa capacité à traiter efficacement les dépendances entre les mots et les phrases dans un texte, ce qui est essentiel pour identifier les fausses nouvelles qui reposent souvent sur des formulations ambiguës ou trompeuses. D'autre part, l'utilisation de CNN permet d'extraire des caractéristiques locales pertinentes, tandis que l'ajout du modèle HAN améliore encore plus la capacité du système à identifier les informations critiques dans les textes.

Les forces des contributions résident dans plusieurs aspects clés. Tout d'abord, l'approche hybride combinant CNN, BiLSTM et HAN a permis de capturer à la fois des informations locales (via CNN) et globales (via BiLSTM et HAN), tout en appliquant un mécanisme d'attention hiérarchique qui accorde plus de poids aux parties importantes du texte, améliorant ainsi la précision de la détection des fausses nouvelles. Ensuite, cette approche a démontré une grande adaptabilité à différents contextes, en particulier lorsqu'elle a été testée sur des jeux de données variés tels que le Liar Dataset, FakeNewsNet, ainsi que sur des ensembles de données spécifiques à la crise du tremblement de terre au Maroc. Elle a prouvé son efficacité dans des scénarios où la propagation de fausses informations est fréquente, notamment en période de crise. Enfin, l'utilisation de techniques NLP avancées, comme les embeddings de mots Word2Vec et GloVe, a permis une meilleure représentation sémantique des mots dans les textes, renforçant ainsi les performances globales des modèles. L'analyse comparative a montré que la combinaison d'approches basées sur les embeddings et l'attention permet de capturer des nuances linguistiques subtiles, souvent difficiles à détecter avec des modèles plus simples.

Pour répondre aux limites identifiées, plusieurs axes de recherche futurs sont envisagés. Tout d'abord, une collecte plus large et plus diversifiée de jeux de données pourrait renforcer la capacité de généralisation des modèles. À cet effet, une collaboration avec la Société nationale de radiodiffusion et de télévision (SNRT) a déjà été initiée pour obtenir des données spécifiques liées au tremblement de terre, et un travail de collecte de données a commencé en utilisant des technologies de Big Data et de Business Intelligence (BI). L'application de techniques d'apprentissage par transfert pourrait également permettre de mieux exploiter les connaissances acquises sur un ensemble de données et de les appliquer à de

nouveaux contextes avec moins de réentraînement. Par ailleurs, l'optimisation des performances computationnelles reste une priorité afin de rendre ces approches plus viables dans des scénarios de prise de décision en temps réel, notamment en situation de crise. Collaborer avec des organismes de gestion des crises et des entités spécialisées dans la vérification des informations permettrait d'adapter les solutions proposées à des scénarios pratiques et de renforcer leur impact sociétal.

Les contributions présentées dans ce chapitre apportent des avancées notables à la détection des fausses nouvelles en ligne, en exploitant des modèles d'apprentissage profond robustes et adaptables. En particulier, l'utilisation combinée de CNN, BiLSTM et HAN a démontré son efficacité dans la capture des relations locales et globales dans les textes. Bien que les résultats soient prometteurs, des efforts supplémentaires sont nécessaires pour améliorer la généralisation et l'efficacité computationnelle de ces modèles, en particulier dans des environnements en évolution rapide comme les crises humanitaires et les événements globaux majeurs.

Conclusion

Les résultats obtenus dans cette étude ont permis de démontrer que l'intégration de techniques avancées de traitement du langage naturel et de modèles d'apprentissage profond améliore considérablement la détection des fausses nouvelles. En particulier, des architectures complexes telles que le CNN, le BiLSTM, et le HAN se sont révélées particulièrement efficaces, chaque modèle contribuant à une meilleure compréhension des aspects contextuels et sémantiques des données textuelles.

Parmi les modèles testés, les architectures pré-entraînées telles que BERT ont montré des performances optimales, surpassant d'autres modèles sur la base des métriques d'évaluation, notamment la précision, le rappel, et le score F1. De plus, les expérimentations sur divers ensembles de données, y compris des articles en ligne et des tweets, ont souligné l'importance d'une approche multimodale et adaptable pour capturer les nombreuses nuances des fausses nouvelles.

Enfin, la contribution majeure de notre travail réside dans l'application de notre modèle de détection des fausses nouvelles à des scénarios réels, tels que la gestion de crises comme le séisme au Maroc en 2023. Cette approche a montré comment des outils basés sur l'intelligence artificielle peuvent être utilisés pour vérifier et trier efficacement les informations critiques en temps de crise, permettant une meilleure gestion de la diffusion de l'information. À l'avenir, nous prévoyons d'étendre cette méthodologie à d'autres domaines tels que la santé publique, les finances, et la sécurité, afin d'assurer une robustesse accrue de notre modèle.

L'intégration de techniques d'apprentissage par transfert et l'utilisation de données en temps réel amélioreront encore la capacité du modèle à détecter et à traiter les fausses nouvelles dans des environnements dynamiques et en constante évolution.

Conclusion Générale et Perspectives

Dans cette thèse, nous avons développé et évalué diverses approches novatrices pour la détection des fausses nouvelles en nous appuyant sur des techniques avancées d'apprentissage profond et de traitement du langage naturel. L'objectif principal de notre étude était de transformer le problème de la détection de fausses informations en un problème de classification textuelle, ce qui nous a permis de tester différentes architectures de réseaux neuronaux et combinaisons de techniques de prétraitement des données afin d'optimiser à la fois la précision et la robustesse des systèmes de détection.

Nous avons consacré une attention particulière à l'exploration de modèles basés sur CNN et Bi-LSTM pour capturer les relations locales et contextuelles dans les données textuelles. En complément, nous avons étudié des approches plus complexes, comme le modèle HAN, qui se distingue par sa capacité à pondérer de manière dynamique l'importance des mots et des phrases dans le processus de classification. Ces modèles ont permis d'extraire des motifs textuels récurrents et des dépendances séquentielles à travers les documents.

Notre étude comparative[7],[8] entre ces différentes architectures a permis de souligner non seulement les forces et les faiblesses de chaque modèle, mais aussi d'identifier les meilleures combinaisons possibles de méthodes d'extraction de caractéristiques et de réseaux neuronaux. En testant ces modèles sur plusieurs ensembles de données, nous avons obtenu des résultats concluants qui démontrent l'efficacité des réseaux neuronaux profonds, notamment les modèles hybrides comme CNN-BiLSTM et HAN, pour améliorer les performances de détection des fausses nouvelles dans des contextes variés. Nous avons également intégré des architectures pré-entraînées comme BERT, dont l'efficacité pour des tâches de NLP est reconnue. BERT, avec sa capacité à saisir les nuances sémantiques profondes grâce à son apprentissage bidirectionnel, s'est révélé être un modèle de référence performant pour détecter les fausses nouvelles. Sa force réside dans sa compréhension contextuelle des mots dans une phrase, ce qui permet d'améliorer la précision globale de classification.

Cette diversité d'approches et d'architectures que nous avons étudiées [16],[17],[63],[69],[84],[225] a mis en lumière les spécificités et les défis de la détection des fausses nouvelles par rapport à d'autres tâches de classification textuelle. Notre travail jette ainsi les bases pour de futures recherches et applications dans ce domaine, en visant à affiner davantage les systèmes de détection pour qu'ils soient plus adaptés aux besoins de la société actuelle, où la désinformation prolifère à un rythme alarmant.

Nos résultats ont clairement montré que les modèles basés sur BERT, ainsi que les approches hybrides combinant CNN et Bi-LSTM, ont surpassé les autres architectures en termes de précision, de capacité de généralisation et de robustesse, lorsqu'ils ont été appliqués à divers ensembles de données, y compris

l'ensemble ISOT [234] et d'autres sources hétérogènes[31],[198]. Ces modèles se sont distingués non seulement par leur capacité à traiter des volumes importants de données textuelles, mais aussi par leur aptitude à capter les nuances contextuelles et sémantiques des textes, ce qui est crucial dans la détection des fausses nouvelles.

L'intégration de techniques d'extraction de caractéristiques telles que le BoW, le TF-IDF et les embeddings de mots (Word2Vec, GloVe, FastText) a joué un rôle essentiel dans l'amélioration des performances des modèles. Ces méthodes ont permis de capturer à la fois la richesse lexicale et sémantique des documents, en offrant des représentations qui reflètent les relations entre les mots et leur importance dans le texte. Cela a considérablement aidé les modèles à distinguer les contenus authentiques des contenus trompeurs, en identifiant les motifs subtils caractéristiques des fausses nouvelles.

En particulier, BERT, avec sa capacité à comprendre les dépendances contextuelles à plusieurs niveaux dans une phrase, s'est révélé particulièrement efficace pour saisir les subtilités linguistiques qui échappent souvent aux modèles plus simples. De plus, l'approche hybride combinant CNN pour l'extraction de motifs locaux et Bi-LSTM pour la capture des dépendances séquentielles à long terme a permis d'atteindre un équilibre optimal entre précision et complexité du modèle. Cette synergie entre extraction de caractéristiques locales et globales a maximisé les capacités des modèles à gérer des types de données variés, tout en maintenant une grande précision dans la classification.

En somme, nos expériences démontrent l'importance de la combinaison de méthodes avancées d'extraction de caractéristiques avec des architectures de réseaux neuronaux profonds, et soulignent le potentiel des approches hybrides dans la détection des fausses nouvelles, un domaine où la richesse sémantique et contextuelle des données joue un rôle clé.

Notre contribution principale dans cette recherche réside dans le développement et l'introduction d'une approche novatrice combinant les architectures CNN, Bi-LSTM et HAN, spécifiquement conçue pour améliorer la détection des fausses nouvelles, en particulier dans des contextes de crise. Ce modèle se distingue par son approche synergique : il exploite les avantages de chaque architecture pour traiter efficacement les complexités linguistiques et contextuelles des informations trompeuses.

Le modèle utilise CNN pour extraire des motifs locaux à partir des données textuelles, notamment les relations immédiates entre les mots, ce qui permet d'identifier des indices subtils au sein des phrases. Ensuite, un modèle Bi-LSTM est utilisé pour capturer les dépendances séquentielles à long terme, en tenant compte du contexte global des documents, tant dans le sens direct que dans le sens inverse. Cette capacité à modéliser les relations à longue portée améliore considérablement la compréhension du contenu narratif et contextuel du texte.

L'élément clé de cette architecture est l'intégration d'un HAN, qui apporte une dimension supplémentaire d'efficacité en hiérarchisant les mots et les phrases les plus pertinents au sein d'un document. Le HAN permet non seulement de pondérer l'importance relative des différentes unités textuelles, mais il fournit également au modèle la capacité de se concentrer sur les éléments cruciaux qui signalent la tromperie, tels que les changements de ton, les contradictions ou les affirmations sans fondement. Cette hiérarchisation dynamique des informations améliore considérablement la capacité du modèle à capturer

les motifs linguistiques complexes et les subtilités contextuelles caractéristiques des fausses nouvelles.

Grâce à cette combinaison puissante de techniques, notre modèle est particulièrement bien adapté aux situations de crise, où la rapidité et la précision de la détection des fausses informations sont essentielles. En hiérarchisant intelligemment les informations, notre approche permet non seulement de renforcer la précision de la détection, mais aussi de s'adapter aux évolutions rapides des données en temps réel. Cela en fait une solution robuste et flexible, capable de relever les défis posés par la diffusion de fausses nouvelles dans des environnements dynamiques et imprévisibles.

Cependant, nos travaux ont également mis en lumière plusieurs limites cruciales qu'il convient de considérer. La première concerne la difficulté de maintenir une précision élevée dans un environnement où l'information évolue rapidement, notamment en période de crise. Les nouvelles informations sont souvent contradictoires, partielles, voire erronées, ce qui complique considérablement la tâche du modèle. Cette évolution rapide de l'information exige que le modèle soit capable de s'adapter presque en temps réel, un défi technique et computationnel important.

Un autre obstacle majeur réside dans la disponibilité et la fiabilité des données en temps de crise. Les données sont issues de différentes sources, comme les réseaux sociaux, les articles de presse ou encore les publications informelles, où la qualité et la véracité des informations peuvent être très différentes. Cela rend difficile la construction d'un ensemble de données de référence exhaustif et représentatif, nécessaire pour garantir une bonne généralisation du modèle. Par ailleurs, la grande diversité des styles linguistiques et des formats de communication dans ces sources rend la tâche encore plus complexe. Les nuances contextuelles et les variations culturelles peuvent échapper au modèle, réduisant ainsi son efficacité.

Enfin, l'adaptabilité du modèle à des situations imprévues et à des domaines différents est également une contrainte. Bien que notre approche repose sur des techniques avancées comme le CNN, le Bi-LSTM et le HAN, l'efficacité du modèle dépend fortement des données sur lesquelles il a été entraîné. La nécessité d'entraîner le modèle sur des ensembles de données vastes et diversifiés pour assurer sa robustesse face aux nouvelles formes de fausses informations demeure un défi central.

Ces limites soulignent l'importance de poursuivre les recherches pour améliorer non seulement la rapidité et la précision du modèle, mais aussi sa capacité à s'adapter à l'évolution dynamique de l'information dans des environnements critiques.

Pour surmonter les limites identifiées et renforcer notre modèle de détection des fausses nouvelles, plusieurs pistes de recherche sont envisagées. Tout d'abord, l'intégration de sources de données supplémentaires apparaît comme un axe central. En élargissant notre modèle au-delà des données textuelles pour inclure des informations multimédias telles que les images, vidéos et fichiers audio, il serait possible de traiter des formes plus complexes de désinformation, souvent présentes dans les réseaux sociaux. La capacité à combiner plusieurs modalités de données renforcerait ainsi la robustesse et l'efficacité du modèle dans des environnements où les informations trompeuses sont diffusées sous diverses formes.

Ensuite, l'application de techniques d'apprentissage par transfert représente un domaine prometteur.

En exploitant des modèles pré-entraînés sur de vastes ensembles de données spécifiques à certains domaines (comme les médias sociaux, la politique ou les crises sanitaires), nous pourrions améliorer la généralisation de notre modèle à des types d'informations ou des contextes diversifiés. Cela entraînerait non seulement une diminution du temps de formation, mais également une capacité du modèle à s'adapter à de nouvelles formes de fausses nouvelles, même dans des contextes pour lesquels il n'a pas été spécifiquement entraîné.

De plus, l'une des évolutions majeures envisagées concerne l'amélioration des mécanismes d'attention utilisés dans le modèle. L'introduction de nouvelles méthodes d'attention plus fines, capables de pondérer de manière plus intelligente les segments les plus critiques du texte ou des éléments multimédias, pourrait affiner encore davantage la capacité du modèle à détecter des motifs linguistiques complexes et cachés.

Un autre axe de recherche concerne l'amélioration des performances computationnelles du modèle. L'utilisation de techniques de compression de modèles et de pruning pourrait rendre notre modèle plus léger et plus rapide, ce qui est essentiel dans des situations de crise où le traitement de l'information en temps réel est crucial. Cette amélioration permettrait d'intégrer le modèle dans des systèmes embarqués ou des plateformes en ligne, où les ressources matérielles sont souvent limitées.

Enfin, un aspect clé des futurs travaux sera l'évaluation continue du modèle dans des scénarios réels, notamment en collaboration avec des organisations spécialisées dans la gestion des crises ou la lutte contre la désinformation. Cette évaluation sur le terrain fournirait des retours précieux sur l'efficacité du modèle dans des environnements dynamiques et non contrôlés, conduisant à des ajustements et améliorations supplémentaires. Par exemple, des tests sur des données liées aux catastrophes naturelles, comme les séismes, ou à des événements politiques, permettraient d'évaluer la robustesse du modèle face aux pressions du monde réel.

Les futurs travaux viseront à enrichir notre modèle en incorporant des données multimodales, en exploitant des modèles pré-entraînés pour la généralisation[235], et en améliorant les performances computationnelles pour une application rapide dans des situations critiques. Ces avancées assureront une détection plus précise, efficace et adaptée à la nature complexe et en constante évolution de la désinformation.

Publications

Cette section présente les différentes contributions dans des revues internationales et des conférences internationales. Les articles publiés dans des revues internationales sont le fruit des recherches présentées dans cette thèse. Certaines des conférences internationales abordent des articles directement liés à cette thèse, tandis que d'autres articles ont été publiés pendant la préparation de cette thèse dans le cadre d'autres projets.

Revues Internationales

Nos contributions dans les revues internationales sont les suivantes :

— **Ennejjai, I.**, Ariss, A., Mabrouki, J., Fouad, Y., Alabdultif, A., Chaganti, R., ... & Ziti, S. (2024). An Artificial intelligence Approach to Fake News Detection in the Context of the Morocco Earthquake. Data and Metadata, 3, 377-377.

— **Ennejjai, I.**, Ariss, A., Mabrouki, J., & Ziti, S. (2024). Enhancing Misinformation Detection Using Long Short-Term Memory (LSTM) and Bidirectional LSTM (BiLSTM) with Word Embedding Techniques. Discrete Mathematics, Algorithms and Applications.

— Ariss, A.,**Ennejjai, I.**, Mabrouki, J., Lamjid, A., Kharmoum, N., & Ziti, S. (2024). Tracking System for Living Beings and Objects : Integration of Accessible Mathematical Contributions and Graph Theory in Tracking System Design. Data and Metadata, 3, 376-376.

— Lamjid, A., Anass, A.,**Ennejjai, I.**, Mabrouki, J., & Soumia, Z. (2024). Enhancing the hiring process : A predictive system for soft skills assessment. Data and Metadata, 3, 387-387.

Conférences Internationales et Livres

Nos contributions dans les conférences internationales sont les suivantes :

— Lamjid, A., Ariss, **Ennejjai, I.**, Mabrouki, J., Lamzouri, F. Z., & Ziti, S. (2024). Predictive Hiring Micro Systems for Data Analysts Through Soft Skills Assessment. In Technical and Technological Solutions Towards a Sustainable Society and Circular Economy (pp. 295-302). Cham : Springer Nature Switzerland.

— Benchrifa, M., Mabrouki, J., Ariss, A., **Ennejjai, I.**, Hmouni, D., & El-Moustaqim, K. (2024). Study of the Physicochemical Properties of Lavender Essential Oil (Lavandula Stoechas). In Advanced Systems for Environmental Monitoring, IoT and the application of Artificial Intelligence (pp. 329-341). Cham : Springer Nature Switzerland.

— **Ennejjai, I.**, Ariss, A., Kharmoum, N., Rhalem, W., Ziti, S., & Ezziyyani, M. (2022, May). Artificial intelligence for fake news. In International Conference on Advanced Intelligent Systems for Sustainable Development (pp. 77-91). Cham : Springer Nature Switzerland.

— Ariss, A.,**Ennejjai, I.**, Kharmoum, N., Rhalem, W., Ziti, S., & Ezziyyani, M. (2022, May). Tracking Methods : Comprehensive Vision and Multiple Approaches. In International Conference on Advanced Intelligent Systems for Sustainable Development (pp. 40-54). Cham : Springer Nature Switzerland.

Bibliographie

[1] Priyanka Meel and Dinesh Kumar Vishwakarma. Fake news, rumor, information pollution in social media and web : A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems with Applications*, 153 :112986, 2020.

[2] Francois Chollet. *Deep learning with Python*. Simon and Schuster, 2021.

[3] Kumar Ajitesh. Different types of cnn architectures explained : Examples. `https://vitalflux.com/different-types-of-cnn-architectures-explained-examples/`. Accessed on 6 January 2024.

[4] Christopher Olah et al. Understanding lstm networks. 2015.

[5] Y. Verma. Complete guide to bidirectional lstm (with python codes). Available online : `https://analyticsindiamag.com/complete-guide-to-bidirectional-lstm-with-python-codes/` (accessed on 6 January 2024), 2021.

[6] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics : human language technologies*, pages 1480–1489, 2016.

[7] I. Ennejjai, S. I. El Ahrache, and B. Hassan. Fake news detection using deep learning. In *8th International Conference on Innovation and New Trends in Information Technology (INNOVATIONS)*, pages 1–8, 2020.

[8] Imane Ennejjai, Anass Ariss, Nassim Kharmoum, Wajih Rhalem, Soumia Ziti, and Mostafa Ezziyyani. Artificial intelligence for fake news. In *International Conference on Advanced Intelligent Systems for Sustainable Development*, pages 77–91. Springer, 2022.

[9] ITU Statistics. Available online : https ://www. itu. int/en/itu-d/statistics. *Pages/stat/default. aspx (accessed on 30 June 2021)*, 2020.

[10] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media : A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1) :22–36, 2017.

[11] William Scott Paka, Rachit Bansal, Abhay Kaushik, Shubhashis Sengupta, and Tanmoy Chakraborty. Cross-sean : A cross-stitch semi-supervised neural attention model for covid-19 fake news detection. *Applied Soft Computing*, 107 :107393, 2021.

[12] Elio Masciari, Vincenzo Moscato, Antonio Picariello, and Giancarlo Sperlí. Detecting fake news by image analysis. In *Proceedings of the 24th symposium on international database engineering & Applications*, pages 1–5, 2020.

[13] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2) :211–236, 2017.

[14] Galit Shmueli, Peter C Bruce, Inbal Yahav, Nitin R Patel, and Kenneth C Lichtendahl Jr. *Data mining for business analytics : concepts, techniques, and applications in R*. John Wiley & Sons, 2017.

[15] Imane Ennejjai, Anass Ariss, Jamal Mabrouki, and Soumia Ziti. Enhancing misinformation detection using long short-term memory (lstm) and bidirectional lstm (bilstm) with word embedding techniques. *Discrete Mathematics, Algorithms and Applications*, 2024.

[16] Pummy Dhiman, Amandeep Kaur, Celestine Iwendi, and Senthil Kumar Mohan. A scientometric analysis of deep learning approaches for detecting fake news. *Electronics*, 12(4) :948, 2023.

[17] Rubab Roshan, Irfan Ali Bhacho, and Sammer Zai. Comparative analysis of tf–idf and hashing vectorizer for fake news detection in sindhi : A machine learning and deep learning approach. *Engineering Proceedings*, 46(1) :5, 2023.

[18] Imane Ennejjai, Anass Ariss, Jamal Mabrouki, Yasser Fouad, Abdulatif Alabdultif, Rajasekhar Chaganti, Karima Salah Eddine, Asmaa Lamjid, and Soumia Ziti. An artificial intelligence approach to fake news detection in the context of the morocco earthquake ; [un enfoque de inteligencia artificial para la detección de noticias falsas en el contexto del terremoto de marruecos]. *Data and Metadata*, 3, 2024. Cited by : 1 ; All Open Access, Hybrid Gold Open Access.

[19] Md Rafiqul Islam, Shaowu Liu, Xianzhi Wang, and Guandong Xu. Deep learning for misinformation detection on online social networks : a survey and new perspectives. *Social Network Analysis and Mining*, 10(1) :82, 2020.

[20] Sarah A Alkhodair, Steven HH Ding, Benjamin CM Fung, and Junqiang Liu. Detecting breaking news rumors of emerging topics in social media. *Information Processing & Management*, 57(2) :102018, 2020.

[21] Bhavtosh Rath, Wei Gao, Jing Ma, and Jaideep Srivastava. Utilizing computational trust to identify rumor spreaders on twitter. *Social Network Analysis and Mining*, 8(1) :64, 2018.

[22] Abeer Aldayel and Walid Magdy. Your stance is exposed ! analysing possible factors for stance detection on social media. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW) :1–20, 2019.

[23] Sawinder Kaur, Parteek Kumar, and Ponnurangam Kumaraguru. Automating fake news detection system using multi-level voting model. *Soft Computing*, 24(12) :9049–9069, 2020.

[24] L John Martin. Disinformation : An instrumentality in the propaganda arsenal. *Political Communication*, 2(1) :47–64, 1982.

[25] Ladislav Bittman. The use of disinformation by democracies. *International Journal of Intelligence and Counter Intelligence*, 4(2) :243–261, 1990.

[26] Claire Wardle et al. Fake news. it's complicated. *First draft*, 16 :1–11, 2017.

[27] Martin Kragh and Sebastian Åsberg. Russia's strategy for influence through public diplomacy and active measures : the swedish case. *Journal of Strategic Studies*, 40(6) :773–816, 2017.

[28] Don Fallis. What is disinformation ? *Library trends*, 63(3) :401–426, 2015.

[29] Claire Wardle and Hossein Derakhshan. Information disorder : Toward an interdisciplinary framework for research and policy making. `https://rm.coe.int/information-disorder-toward-an-interdisciplinary-framework-forresearc/168076277c`, 2018. Accessed : 2024-08-02.

[30] Leonie Haiden, Chris McManus, Céline Michaud, Emma Duffy, Kulsoom Ranautta-Sambhi, and Claire Ilbury. *A Roadmap*. NATO Strategic Communications Centre of Excellence, 2018.

[31] William Yang Wang. " liar, liar pants on fire" : A new benchmark dataset for fake news detection. *arXiv preprint arXiv :1705.00648*, 2017.

[32] Sheng-Xuan Lin, Bo-Yi Wu, Tzu-Hsuan Chou, Ying-Jia Lin, and Hung-Yu Kao. Bidirectional perspective with topic information for stance detection. In *2020 International Conference on Pervasive Artificial Intelligence (ICPAI)*, pages 1–8. IEEE, 2020.

[33] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi : A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 797–806, 2017.

[34] Samir Bajaj. The pope has a new baby ! *Fake news detection using deep learning*, pages 1–8, 2017.

[35] Eugenio Tacchini, Gabriele Ballarin, Marco L Della Vedova, Stefano Moret, and Luca De Alfaro. Some like it hoax : Automated fake news detection in social networks. *arXiv preprint arXiv :1704.07506*, 2017.

[36] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detecting opinion spams and fake news using text classification. *Security and Privacy*, 1(1) :e9, 2018.

[37] Aditi Gupta, Ponnurangam Kumaraguru, Carlos Castillo, and Patrick Meier. Tweetcred : Real-time credibility assessment of content on twitter. In *Social Informatics : 6th International Conference, SocInfo 2014, Barcelona, Spain, November 11-13, 2014. Proceedings 6*, pages 228–243. Springer, 2014.

[38] Svitlana Volkova, Kyle Shaffer, Jin Yea Jang, and Nathan Hodas. Separating facts from fiction : Linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, pages 647–653, 2017.

[39] Wei Wei and Xiaojun Wan. Learning to identify ambiguous and misleading news headlines. *arXiv preprint arXiv :1705.06031*, 2017.

[40] Behzad Tabibian, Isabel Valera, Mehrdad Farajtabar, Le Song, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. Distilling information reliability and source trustworthiness from digital traces. In *Proceedings of the 26th International Conference on World Wide Web*, pages 847–855, 2017.

[41] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. Computational fact checking from knowledge networks. *PloS one*, 10(6) :e0128193, 2015.

[42] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684, 2011.

[43] Sejeong Kwon, Meeyoung Cha, and Kyomin Jung. Rumor detection over varying time windows. *PloS one*, 12(1) :e0168344, 2017.

[44] Shu Wu, Qiang Liu, Yong Liu, Liang Wang, and Tieniu Tan. Information credibility evaluation on social media. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[45] Mengchen Liu, Liu Jiang, Junlin Liu, Xiting Wang, Jun Zhu, and Shixia Liu. Improving learning-from-crowds through expert validation. In *IJCAI*, pages 2329–2336, 2017.

[46] Sebastian Tschiatschek, Adish Singla, Manuel Gomez Rodriguez, Arpit Merchant, and Andreas Krause. Fake news detection in social networks via crowd signals. In *Companion proceedings of the the web conference 2018*, pages 517–524, 2018.

[47] Yang Liu and Yi-Fang Wu. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[48] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting rumors from microblogs with recurrent neural networks. 2016.

[49] Sneha Singhania, Nigel Fernandez, and Shrisha Rao. 3han : A deep neural network for fake news detection. In *Neural Information Processing : 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Proceedings, Part II 24*, pages 572–581. Springer, 2017.

[50] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. Declare : Debunking fake news and false claims using evidence-aware deep learning. *arXiv preprint arXiv :1809.06416*, 2018.

[51] MARINOS ZAGKOTSIS. Fake news detection using deep learning and machine learning methods. 2019.

[52] Μαρίνος Παύλου Ζαγκότσης. *Fake News Detection using Deep Learning and Machine Learning Methods-A comparative study on short and long texts*. PhD thesis, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2019.

[53] Ali Bou Nassif, Ashraf Elnagar, Omar Elgendy, and Yaman Afadar. Arabic fake news detection based on deep contextualized embedding models. *Neural Computing and Applications*, 34(18) :16019–16032, 2022.

[54] Mohamed Osman Hegazi, Yasser Al-Dossari, Abdullah Al-Yahy, Abdulaziz Al-Sumari, and Anwer Hilal. Preprocessing arabic text on social media. *Heliyon*, 7(2), 2021.

[55] Wesam Shishah. Jointbert for detecting arabic fake news. *IEEE Access*, 10 :71951–71960, 2022.

[56] SHAYMAA E Sorour and HANAN E Abdelkader. Afnd : Arabic fake news detection with an ensemble deep cnn-lstm model. *J. Theor. Appl. Inf. Technol*, 100(14) :5072–5086, 2022.

[57] Samah M Alzanin, Aqil M Azmi, and Hatim A Aboalsamh. Short text classification for arabic social media tweets. *Journal of King Saud University-Computer and Information Sciences*, 34(9) :6595–6604, 2022.

[58] Sebastian Kula, Michał Choraś, Rafał Kozik, Paweł Ksieniewicz, and Michał Woźniak. Sentiment analysis for fake news detection by means of neural networks. In *Computational Science–ICCS 2020 : 20th International Conference, Amsterdam, The Netherlands, June 3–5, 2020, Proceedings, Part IV 20*, pages 653–666. Springer, 2020.

[59] I Kadek Sastrawan, I Putu Agung Bayupati, and Dewa Made Sri Arsa. Detection of fake news using deep learning cnn–rnn based methods. *ICT express*, 8(3) :396–408, 2022.

[60] Iftikhar Ahmad, Muhammad Yousaf, Suhail Yousaf, and Muhammad Ovais Ahmad. Fake news detection using machine learning ensemble methods. *Complexity*, 2020 :1–11, 2020.

[61] Benjamin Horne and Sibel Adali. This just in : Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 759–766, 2017.

[62] Xinyi Zhou and Reza Zafarani. A survey of fake news : Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5) :1–40, 2020.

[63] Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu. Unsupervised fake news detection on social media : A generative approach. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5644–5651, 2019.

[64] Dun Li, Haimei Guo, Zhenfei Wang, and Zhiyun Zheng. Unsupervised fake news detection based on autoencoder. *IEEE access*, 9 :29356–29365, 2021.

[65] Adrien Benamira, Benjamin Devillers, Etienne Lesot, Ayush K Ray, Manal Saadi, and Fragkiskos D Malliaros. Semi-supervised learning and graph neural networks for fake news detection. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 568–569, 2019.

[66] Md Saqib Hasan, Rukshar Alam, and Muhammad Abdullah Adnan. Truth or lie : Pre-emptive detection of fake news in different languages through entropy-based active learning and multimodel neural ensemble. In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 55–59. IEEE, 2020.

[67] Yaqing Wang, Weifeng Yang, Fenglong Ma, Jin Xu, Bin Zhong, Qiang Deng, and Jing Gao. Weak supervision for fake news detection via reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 516–523, 2020.

[68] Martin Sarnovskỳ, Viera Maslej-Krešňáková, and Nikola Hrabovská. Annotated dataset for the fake news classification in slovak language. In *2020 18th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, pages 574–579. IEEE, 2020.

[69] Pavel Přibáň, Tomáš Hercig, and Josef Steinberger. Machine learning approach to fact-checking in west slavic languages. In *Proceedings of the international conference on recent advances in natural language processing (RANLP 2019)*, pages 973–979, 2019.

[70] Abhinav Kumar, Sunil Saumya, and Jyoti Prakash Singh. Nitp-ai-nlp@ urdufake-fire2020 : Multilayer dense neural network for fake news detection in urdu news articles. In *FIRE (Working Notes)*, pages 458–463, 2020.

[71] Maaz Amjad, Grigori Sidorov, and Alisa Zhila. Data augmentation using machine translation for fake news detection in the urdu language. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2537–2542, 2020.

[72] Maaz Amjad, Grigori Sidorov, Alisa Zhila, Alexander Gelbukh, and Paolo Rosso. Urdufake@ fire2020 : shared track on fake news identification in urdu. In *Proceedings of the 12th Annual Meeting of the Forum for Information Retrieval Evaluation*, pages 37–40, 2020.

[73] Abhishek Verma, Vanshika Mittal, and Suma Dawn. Find : Fake information and news detections using deep learning. In *2019 twelfth international conference on contemporary computing (IC3)*, pages 1–7. IEEE, 2019.

[74] Alexandros Zervopoulos, Aikaterini Georgia Alvanou, Konstantinos Bezas, Asterios Papamichail, Manolis Maragoudakis, and Katia Kermanidis. Hong kong protests : using natural language processing for fake news detection on twitter. In *Artificial Intelligence Applications and Innovations : 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part II 16*, pages 408–419. Springer, 2020.

[75] Xishuang Dong, Uboho Victor, and Lijun Qian. Two-path deep semisupervised learning for timely fake news detection. *IEEE Transactions on Computational Social Systems*, 7(6) :1386–1398, 2020.

[76] Arkaitz Zubiaga, Elena Kochkina, Maria Liakata, Rob Procter, Michal Lukasik, Kalina Bontcheva, Trevor Cohn, and Isabelle Augenstein. Discourse-aware rumour stance classification in social media using sequential classifiers. *Information Processing & Management*, 54(2) :273–290, 2018.

[77] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *science*, 359(6380) :1146–1151, 2018.

[78] Julio CS Reis, André Correia, Fabrício Murai, Adriano Veloso, and Fabrício Benevenuto. Supervised learning for fake news detection. *IEEE Intelligent Systems*, 34(2) :76–81, 2019.

[79] Shlok Gilda. Notice of violation of ieee publication principles : Evaluating machine learning algorithms for fake news detection. In *2017 IEEE 15th student conference on research and development (SCOReD)*, pages 110–115. IEEE, 2017.

[80] Georgios Gravanis, Athena Vakali, Konstantinos Diamantaras, and Panagiotis Karadais. Behind the cues : A benchmarking study for fake news detection. *Expert Systems with Applications*, 128 :201–213, 2019.

[81] Amila Silva, Ling Luo, Shanika Karunasekera, and Christopher Leckie. Embracing domain differences in fake news : Cross-domain fake news detection using multi-modal data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 557–565, 2021.

[82] Chandra Mouli Madhav Kotteti, Xishuang Dong, Na Li, and Lijun Qian. Fake news detection enhancement with data imputation. In *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, pages 187–192. IEEE, 2018.

[83] Arvinder Pal Singh Bali, Mexson Fernandes, Sourabh Choubey, and Mahima Goel. Comparative performance of machine learning algorithms for fake news detection. In *Advances in Computing and Data Sciences : Third International Conference, ICACDS 2019, Ghaziabad, India, April 12–13, 2019, Revised Selected Papers, Part II 3*, pages 420–430. Springer, 2019.

[84] Arush Agarwal and Akhil Dixit. Fake news detection : an ensemble learning approach. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 1178–1183. IEEE, 2020.

[85] Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. Fndnet–a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61 :32–44, 2020.

[86] Junxiao Xue, Yabo Wang, Shuning Xu, Lei Shi, Lin Wei, and Huawei Song. Mvfnn : Multi-vision fusion neural network for fake news picture detection. In *Computer Animation and Social Agents : 33rd International Conference on Computer Animation and Social Agents, CASA 2020, Bournemouth, UK, October 13-15, 2020, Proceedings 33*, pages 112–119. Springer, 2020.

[87] Anoud Bani-Hani, Oluwasegun Adedugbe, Elhadj Benkhelifa, Munir Majdalawieh, and Feras Al-Obeidat. A semantic model for context-based fake news detection on social media. In *2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–7. IEEE, 2020.

[88] Marion Meyers, Gerhard Weiss, and Gerasimos Spanakis. Fake news detection on twitter using propagation structures. In *Disinformation in Open Online Media : Second Multidisciplinary International Symposium, MISDOOM 2020, Leiden, The Netherlands, October 26–27, 2020, Proceedings 2*, pages 138–158. Springer, 2020.

[89] Maria Nefeli Nikiforos, Spiridon Vergis, Andreana Stylidou, Nikolaos Augoustis, Katia Lida Kermanidis, and Manolis Maragoudakis. Fake news detection regarding the hong kong events from tweets. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 177–186. Springer, 2020.

[90] Duc Minh Nguyen, Tien Huu Do, Robert Calderbank, and Nikos Deligiannis. Fake news detection using deep markov random fields. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1391–1400, 2019.

[91] Reza Mansouri, Mahmood Naderan-Tahan, and Mohammad Javad Rashti. A semi-supervised learning method for fake news detection in social media. In *2020 28th Iranian Conference on Electrical Engineering (ICEE)*, pages 1–5. IEEE, 2020.

[92] Alessandro Bondielli and Francesco Marcelloni. A survey on fake news and rumour detection techniques. *Information sciences*, 497 :38–55, 2019.

[93] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detection of online fake news using n-gram analysis and machine learning techniques. In *Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments : First International Conference, ISDDC 2017, Vancouver, BC, Canada, October 26-28, 2017, Proceedings 1*, pages 127–138. Springer, 2017.

[94] N Smitha and R Bharath. Performance comparison of machine learning classifiers for fake news detection. In *2020 Second international conference on inventive research in computing applications (ICIRCA)*, pages 696–700. IEEE, 2020.

[95] Shengsheng Qian, Jinguang Wang, Jun Hu, Quan Fang, and Changsheng Xu. Hierarchical multi-modal contextual attention network for fake news detection. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 153–162, 2021.

[96] Karishnu Poddar, KS Umadevi, et al. Comparison of various machine learning models for accurate detection of fake news. In *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, volume 1, pages 1–5. IEEE, 2019.

[97] Fantahun Bogale Gereme and William Zhu. Fighting fake news using deep learning : Pre-trained word embeddings and the embedding layer investigated. In *Proceedings of the 2020 3rd International Conference on Computational Intelligence and Intelligent Systems*, pages 24–29, 2020.

[98] Harita Reddy, Namratha Raj, Manali Gala, and Annappa Basava. Text-mining-based fake news detection using ensemble methods. *International Journal of Automation and Computing*, 17(2) :210–221, 2020.

[99] Adrian MP Braşoveanu and Răzvan Andonie. Semantic fake news detection : a machine learning perspective. In *Advances in Computational Intelligence : 15th International Work-Conference on Artificial Neural Networks, IWANN 2019, Gran Canaria, Spain, June 12-14, 2019, Proceedings, Part I 15*, pages 656–667. Springer, 2019.

[100] Arjun Roy, Kingshuk Basak, Asif Ekbal, and Pushpak Bhattacharyya. A deep ensemble framework for fake news detection and classification. *arXiv preprint arXiv :1811.04670*, 2018.

[101] Pritika Bahad, Preeti Saxena, and Raj Kamal. Fake news detection using bi-directional lstm-recurrent neural network. *Procedia Computer Science*, 165 :74–82, 2019.

[102] Ayat Abedalla, Aisha Al-Sadi, and Malak Abdullah. A closer look at fake news detection : A deep learning perspective. In *Proceedings of the 3rd International Conference on Advances in Artificial Intelligence*, pages 24–28, 2019.

[103] Wissam Antoun, Fady Baly, Rim Achour, Amir Hussein, and Hazem Hajj. State of the art models for fake news detection tasks. In *2020 IEEE international conference on informatics, IoT, and enabling technologies (ICIoT)*, pages 519–524. IEEE, 2020.

[104] Zellig S Harris. Distributional structure. *Word*, 10(2-3) :146–162, 1954.

[105] Gerardo Ernesto Rolong Agudelo, Octavio José Salcedo Parra, and Julio Barón Velandia. Raising a model for fake news detection using machine learning in python. In *Challenges and Opportunities in the Digital Era : 17th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2018, Kuwait City, Kuwait, October 30–November 1, 2018, Proceedings 17*, pages 596–604. Springer, 2018.

[106] Vasu Agarwal, H Parveen Sultana, Srijan Malhotra, and Amitrajit Sarkar. Analysis of classifiers for fake news detection. *Procedia Computer Science*, 165 :377–383, 2019.

[107] Paweł Ksieniewicz, Michał Choraś, Rafał Kozik, and Michał Woźniak. Machine learning methods for fake news classification. In *Intelligent Data Engineering and Automated Learning–IDEAL 2019 : 20th International Conference, Manchester, UK, November 14–16, 2019, Proceedings, Part II 20*, pages 332–339. Springer, 2019.

[108] Yin Zhang, Rong Jin, and Zhi-Hua Zhou. Understanding bag-of-words model : a statistical framework. *Int. J. Mach. Learn. Cybern.*, 1(1-4) :43–52, December 2010.

[109] Bashar Al Asaad and Madalina Erascu. A tool for fake news detection. In *2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 379–386. IEEE, 2018.

[110] Julio CS Reis, André Correia, Fabrício Murai, Adriano Veloso, and Fabrício Benevenuto. Explainable machine learning for fake news detection. In *Proceedings of the 10th ACM conference on web science*, pages 17–26, 2019.

[111] Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. Multiclass fake news detection using ensemble machine learning. In *2019 IEEE 9th international conference on advanced computing (IACC)*, pages 103–107. IEEE, 2019.

[112] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

[113] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*, 2013.

[114] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove : Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[115] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5 :135–146, 2017.

[116] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert : Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*, 2018.

[117] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert : smaller, faster, cheaper and lighter. *arXiv preprint arXiv :1910.01108*, 2019.

[118] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta : A robustly optimized bert pretraining approach. *arXiv preprint arXiv :1907.11692*, 2019.

[119] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart : Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv :1910.13461*, 2019.

[120] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra : Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv :2003.10555*, 2020.

[121] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet : Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

[122] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8) :9, 2019.

[123] Antoine Louis et al. Master's thesis : Netbert : A pre-trained language representation model for computer networking. 2020.

[124] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert : A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv :1909.11942*, 2019.

[125] Rohit Kumar Kaliyar. Fake news detection using a deep neural network. In *2018 4th international conference on computing communication and automation (ICCCA)*, pages 1–7. IEEE, 2018.

[126] Ehesas Mia Mahir, Saima Akhter, Mohammad Rezwanul Huq, et al. Detecting fake news using machine learning and deep learning algorithms. In *2019 7th international conference on smart computing & communications (ICSCC)*, pages 1–5. IEEE, 2019.

[127] Gaurav Bhatt, Aman Sharma, Shivam Sharma, Ankush Nagpal, Balasubramanian Raman, and Ankush Mittal. Combining neural, statistical and external features for fake news stance identification. In *Companion Proceedings of the The Web Conference 2018*, pages 1353–1357, 2018.

[128] Aswini Thota, Priyanka Tilak, Simrat Ahluwalia, and Nibrat Lohia. Fake news detection : a deep learning approach. *SMU Data Science Review*, 1(3) :10, 2018.

[129] Muhammad Umer, Zainab Imtiaz, Saleem Ullah, Arif Mehmood, Gyu Sang Choi, and Byung-Won On. Fake news stance detection using deep learning architecture (cnn-lstm). *IEEE Access*, 8 :156695–156706, 2020.

[130] Saarthak Sangamnerkar, R Srinivasan, MR Christhuraj, and Rajeev Sukumaran. An ensemble technique to detect fabricated news article using machine learning and natural language processing techniques. In *2020 International Conference for Emerging Technology (INCET)*, pages 1–7. IEEE, 2020.

[131] Stefan Helmstetter and Heiko Paulheim. Weakly supervised learning for fake news detection on twitter. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 274–277. IEEE, 2018.

[132] Amir Pouran Ben Veyseh, My T Thai, Thien Huu Nguyen, and Dejing Dou. Rumor detection in social networks via deep contextual modeling. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 113–120, 2019.

[133] Sebastian Kula, Michał Choraś, and Rafał Kozik. Application of the bert-based architecture in fake news detection. In *13th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2020) 12*, pages 239–249. Springer, 2021.

[134] Divyam Mehta, Aniket Dwivedi, Arunabha Patra, and M Anand Kumar. A transformer-based architecture for fake news classification. *Social network analysis and mining*, 11 :1–12, 2021.

[135] Sherry Girgis, Eslam Amer, and Mahmoud Gadallah. Deep learning algorithms for detecting fake news in online text. In *2018 13th international conference on computer engineering and systems (ICCES)*, pages 93–97. IEEE, 2018.

[136] Manoj Kumar Balwant. Bidirectional lstm based on pos tags and cnn architecture for fake news detection. In *2019 10th International conference on computing, communication and networking technologies (ICCCNT)*, pages 1–6. IEEE, 2019.

[137] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[138] Vladimir Vapnik. Support-vector networks. *Machine learning*, 20 :273–297, 1995.

[139] Anmol Uppal, Vipul Sachdeva, and Seema Sharma. Fake news detection using discourse segment structure analysis. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 751–756. IEEE, 2020.

[140] Ludmila I Kuncheva. *Combining pattern classifiers : methods and algorithms*. John Wiley & Sons, 2014.

[141] Hnin Ei Wynne and Zar Zar Wint. Content based fake news detection using n-gram models. In *Proceedings of the 21st international conference on information integration and web-based applications & services*, pages 669–673, 2019.

[142] Leo Breiman. Random forests. *Machine learning*, 45 :5–32, 2001.

[143] Yoav Freund, Robert E Schapire, et al. Experiments with a new boosting algorithm. In *icml*, volume 96, pages 148–156. Citeseer, 1996.

[144] Baida Abdulredha Hamdan. Neural network principles and its application. *Webology*, 19(1) :3955–3970, 2022.

[145] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[146] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019.

[147] S. Huang, Y. Xu, X. Liu, and M. Sun. Dual-channel convolutional neural network with attention mechanism for sentiment analysis. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1434–1443, 2018.

[148] X. Qiu, Y. Tang, and Y. Song. A bi-lstm-based recognition system for handwritten character recognition. In *International Conference on Computer Science, Engineering and Applications*, pages 437–444. Springer, 2020.

[149] A. Arora, P. Saxena, and N. Kaur. Fake news detection using part-of-speech based convolutional neural networks. *International Journal of Intelligent Systems and Applications*, 12(6) :1–7, 2020.

[150] Jamal Nasir, Osama Khan, and Iraklis Varlamis. Fake news detection : A hybrid cnn-rnn based deep learning approach. *International Journal of Information Management Data Insights*, 1 :100007, 2021.

[151] Te Han et al. A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults. *Knowledge-based systems*, 165 :474–487, 2019.

[152] Ye Zhang and Byron Wallace. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv :1510.03820*, 2015.

[153] Z. Yang, Y. Zhang, and C. Sun. Detecting fake news headlines with convolutional neural networks. *Information Processing Management*, 57(1) :102200, 2020.

[154] T. Wang, H. Huang, J. Liu, and Y. Yang. Detecting fake reviews with convolutional neural networks and linguistic features. *IEEE Access*, 9 :39532–39542, 2021.

[155] Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[156] X. Ma, Y. Zhou, Z. Li, and M. Lyu. Detect rumors in microblog posts using propagation structure via kernel learning. In *AAAI Conference on Artificial Intelligence*, 2018.

[157] K. Yang and J. Qiu. Multi-channel convolutional neural networks for fake news detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1952–1962, 2019.

[158] W. Wang, X. Wang, Y. Li, and W. Zhou. A hybrid model combining cnn and bi-lstm for fake news detection. *IEEE Access*, 8 :95006–95017, 2020.

[159] S. Ghosh, S. Singhal, and R. R. Shah. Exploring the use of multimodal features and cnns for fake news detection. In *Proceedings of the 2021 6th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–7, 2021.

[160] J. Wang, Y. Li, J. Li, and Y. Li. Fake news detection via hierarchical convolutional neural networks with attention. *IEEE Access*, 9 :54415–54425, 2021.

[161] Aiquan Yuan, Gang Bai, Po Yang, Yanni Guo, and Xinting Zhao. Handwritten english word recognition based on convolutional neural networks. In *2012 international conference on frontiers in handwriting recognition*, pages 207–212. IEEE, 2012.

[162] Attar Ahmed Ali and et al. Linguistic features and bi-lstm for identification of fake news. *Electronics*, 12(13) :2942, 2023.

[163] William Yang Wang. "liar, liar pants on fire" : A new benchmark dataset for fake news detection. *arXiv preprint*, 2017.

[164] Federico Monti and et al. Fake news detection on social media using geometric deep learning. *arXiv preprint*, 2019.

[165] P. Jia, Y. Du, J. Hu, H. Li, X. Li, and X. Chen. An improved bilstm approach for user stance detection based on external commonsense knowledge and environment information. *Applied Sciences*, 12(21) :10968, 2022.

[166] C. Tan, L. Lee, and B. Pang. Bilstm with attention for detecting fake news. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3391–3401. Association for Computational Linguistics, 2019.

[167] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu. Beyond news contents : The role of social context for fake news detection. In *Proceedings of the 2019 World Wide Web Conference*, pages 2495–2501. ACM, 2019.

[168] X. Feng, W. Yu, Z. Ye, and Y. Liu. Fake news detection based on bidirectional lstm with graph attention. *Journal of Intelligent Fuzzy Systems*, 40(2) :1897–1908, 2021.

[169] A. Carrillo, L. F. Cantú, L. Tejerina, and A. Noriega. Individual explanations in machine learning models : A case study on poverty estimation, 2021. arXiv preprint.

[170] K. Ropiak and P. Artiemjew. On a hybridization of deep learning and rough set based granular computing. *Algorithms*, 13(3) :63, 2020.

[171] F. K. De-La-Cruz-Arcela, J. S. Martinez-Castillo, E. Altamirano-Flores, and J. C. Alvarez-Merino. Application of lean manufacturing tools to reduce downtime in a small metalworking facility. In *2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 551–555, 2019.

[172] Feng QIAN and et al. Neural user response generator : Fake news detection with collective user intelligence. In *IJCAI*, pages 3834–3840, 2018.

[173] Y. Chen, J. Xiang, J. Zhang, Y. Zhang, and X. Huang. Multi-task learning based bidirectional lstm for detecting stance and veracity of news. *Knowledge-Based Systems*, 236 :107156, 2022.

[174] Z. Sun, Y. Li, X. Liu, and Y. Sun. Handwritten english word recognition based on deep learning. *Journal of Intelligent Fuzzy Systems*, 35(2) :2397–2406, 2018.

[175] Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. Very deep convolutional networks for text classification. *arXiv preprint arXiv :1606.01781*, 2016.

[176] Yahui Chen. Convolutional neural network for sentence classification. Master's thesis, University of Waterloo, 2015.

[177] Duyu Tang, Bing Qin, and Ting Liu. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1422–1432, 2015.

[178] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv :1412.3555*, 2014.

[179] Xiaoru Wang, Bing Ma, Zhihong Yu, Fu Li, and Yali Cai. Multi-scale decision network with feature fusion and weighting for few-shot learning. *IEEE Access*, 8 :92172–92181, 2020.

[180] Shiwen Ni, Jiawen Li, and Hung-Yu Kao. Mvan : Multi-view attention networks for fake news detection on social media. *IEEE Access*, 9 :106907–106917, 2021.

[181] Lu Yuan, Hangshun Jiang, Hao Shen, Lei Shi, and Nanchang Cheng. Sustainable development of information dissemination : A review of current fake news detection research and practice. *Systems*, 11(9) :458, 2023.

[182] GK Kharate et al. A review : Fake news detection using hierarchical attention network and hypergraph attention. *International Journal of Computing and Digital Systems*, 14(1) :1–xx, 2023.

[183] Jungseok Hong. *Toward Robotic Autonomy in Data-Scarce and Visually Challenging Environments*. PhD thesis, University of Minnesota, 2023.

[184] Yang Wu, Pengwei Zhan, Yunjian Zhang, Liming Wang, and Zhen Xu. Multimodal fusion with co-attention networks for fake news detection. In *Findings of the association for computational linguistics : ACL-IJCNLP 2021*, pages 2560–2569, 2021.

[185] Qian Li, Qingyuan Hu, Youshui Lu, Yue Yang, and Jingxian Cheng. Multi-level word features based on cnn for fake news detection in cultural communication. *Personal and Ubiquitous Computing*, 24 :259–272, 2020.

[186] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[187] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[188] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553) :436–444, 2015.

[189] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 249–256, 2010.

[190] Michael Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015.

[191] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088) :533–536, 1986.

[192] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8) :1735–1780, 1997.

[193] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2) :157–166, 1994.

[194] Boris Hanin. Which neural net architectures give rise to exploding and vanishing gradients ? *Advances in neural information processing systems*, 31, 2018.

[195] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers : Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[196] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[197] PolitiFact. Politifact.com, 2023. Accessed : 2023-07-22.

[198] Parth Patwa, Shivam Sharma, Srinivas Pykl, Vineeth Guptha, Gitanjali Kumari, Md Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. Fighting an infodemic : Covid-19 fake news dataset. In *Combating Online Hostile Posts in Regional Languages during Emergency Situation : First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers 1*, pages 21–29. Springer, 2021.

[199] Yin-Fu Huang and Po-Hong Chen. Fake news detection using an ensemble learning model based on self-adaptive harmony search algorithms. *Expert Systems with Applications*, 159 :113584, 2020.

[200] Yong Fang, Jian Gao, Cheng Huang, Hua Peng, and Runpu Wu. Self multi-head attention-based convolutional neural networks for fake news detection. *PloS one*, 14(9) :e0222713, 2019.

[201] Xishuang Dong, Uboho Victor, Shanta Chowdhury, and Lijun Qian. Deep two-path semi-supervised learning for fake news detection. *arXiv preprint arXiv :1906.05659*, 2019.

[202] Dhruv Khattar, Jaipal Singh Goud, Manish Gupta, and Vasudeva Varma. Mvae : Multimodal variational autoencoder for fake news detection. In *The world wide web conference*, pages 2915–2921, 2019.

[203] Xueyao Zhang, Juan Cao, Xirong Li, Qiang Sheng, Lei Zhong, and Kai Shu. Mining dual emotion for fake news detection. In *Proceedings of the web conference 2021*, pages 3465–3476, 2021.

[204] Lianwei Wu and Yuan Rao. Adaptive interaction fusion networks for fake news detection. In *ECAI 2020*, pages 2220–2227. IOS Press, 2020.

[205] Xinyi Zhou, Jindi Wu, and Reza Zafarani. : Similarity-aware multi-modal fake news detection. In *Pacific-Asia Conference on knowledge discovery and data mining*, pages 354–367. Springer, 2020.

[206] Stephane Schwarz, Antônio Theóphilo, and Anderson Rocha. Emet : Embeddings from multilingual-encoder transformer for fake news detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2777–2781. IEEE, 2020.

[207] Junaed Younus Khan, Md. Tawkat Islam Khondaker, Anindya Iqbal, and Sadia Afroz. A benchmark study on machine learning methods for fake news detection, 05 2019.

[208] John Houvardas and Efstathios Stamatatos. N-gram feature selection for authorship identification. In *International conference on artificial intelligence : Methodology, systems, and applications*, pages 77–86. Springer, 2006.

[209] Gerard Salton. Introduction to modern information retrieval. *McGraw-Hill*, 1983.

[210] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.

[211] Yang Yang, Lei Zheng, Jiawei Zhang, Qingcai Cui, Zhoujun Li, and Philip S Yu. Ti-cnn : Convolutional neural networks for fake news detection. *arXiv preprint arXiv :1806.00749*, 2018.

[212] David Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. *Advances in neural information processing systems*, 24, 2011.

[213] Israel Barrutia-Barreto, Renzo Seminario-Córdova, and Brian Chero-Arana. Fake news detection in internet using deep learning : a review. *Combating Fake News with Computational Intelligence Techniques*, pages 55–67, 2022.

[214] Shlok Gilda. Notice of violation of ieee publication principles : Evaluating machine learning algorithms for fake news detection. In *2017 IEEE 15th Student Conference on Research and Development (SCOReD)*, pages 110–115, 2017.

[215] Maialen Berrondo-Otermin and Antonio Sarasa-Cabezuelo. Application of artificial intelligence techniques to detect fake news : A review. *Electronics*, 12(24) :5041, 2023.

[216] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.

[217] Alex Graves, Marcus Liwicki, Santiago Fernández, Roman Bertolami, Horst Bunke, and Jürgen Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(5) :855–868, 2008.

[218] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv :1409.0473*, 2014.

[219] Ashish Vaswani. Attention is all you need. *arXiv preprint arXiv :1706.03762*, 2017.

[220] Li-Chen Cheng, Yan Tsang Wu, Cheng-Ting Chao, and Jenq-Haur Wang. Detecting fake reviewers from the social context with a graph neural network method. *Decision Support Systems*, 179 :114150, 2024.

[221] Francisco-Javier Rodrigo-Ginés, Jorge Carrillo-de Albornoz, and Laura Plaza. A systematic review on media bias detection : What is media bias, how it is expressed, and how to detect it. *Expert Systems with Applications*, page 121641, 2023.

[222] Selin Gurgun, Deniz Cemiloglu, Emily Arden Close, Keith Phalp, Preslav Nakov, and Raian Ali. Why do we not stand up to misinformation ? factors influencing the likelihood of challenging misinformation on social media and the role of demographics. *Technology in Society*, 76 :102444, 2024.

[223] Zhiguo Qu, Yunyi Meng, Ghulam Muhammad, and Prayag Tiwari. Qmfnd : A quantum multimodal fusion-based fake news detection model for social media. *Information Fusion*, 104 :102172, 2024.

[224] Eduri Raja, Badal Soni, Candy Lalrempuii, and Samir Kumar Borgohain. An adaptive cyclical learning rate based hybrid model for dravidian fake news detection. *Expert Systems with Applications*, 241 :122768, 2024.

[225] Jamal Abdul Nasir, Osama Subhani Khan, and Iraklis Varlamis. Fake news detection : A hybrid cnn-rnn based deep learning approach. *International Journal of Information Management Data Insights*, 1(1) :100007, 2021.

[226] Liang Kuang, Yiting Wang, Tian Hang, Beijing Chen, and Guoying Zhao. A dual-branch neural network for deepfake video detection by detecting spatial and temporal inconsistencies. *Multimedia Tools and Applications*, 81(29) :42591–42606, 2022.

[227] Abdul Samad, Namrata Dhanda, and Rajat Verma. Fake news detection using machine learning. In *International Conference on Artificial Intelligence of Things*, pages 228–243. Springer, 2023.

[228] Ciprian-Octavian Truică and Elena-Simona Apostol. It's all in the embedding ! fake news detection using document embeddings. *Mathematics*, 11(3) :508, 2023.

[229] Dilip Kumar Sharma and Sonal Garg. Ifnd : a benchmark dataset for fake news detection. *Complex & intelligent systems*, 9(3) :2843–2863, 2023.

[230] Yuxiang Wang, Yongheng Zhang, Xuebo Li, and Xinyao Yu. Covid-19 fake news detection using bidirectional encoder representations from transformers based models. *arXiv preprint arXiv :2109.14816*, 2021.

[231] Huosong Xia, Yuan Wang, Justin Zuopeng Zhang, Leven J Zheng, Muhammad Mustafa Kamal, and Varsha Arya. Covid-19 fake news detection : A hybrid cnn-bilstm-am model. *Technological Forecasting and Social Change*, 195 :122746, 2023.

[232] Kate Starbird, Jim Maddock, Mania Orand, Peg Achterman, and Robert M Mason. Rumors, false flags, and digital vigilantes : Misinformation on twitter after the 2013 boston marathon bombing. *IConference 2014 proceedings*, 2014.

[233] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10) :1345–1359, 2009.

[234] ISOT Research Lab. Fake news detection datasets, 2022. Consulté le 23 novembre 2024.

[235] Raghu Raman, Vinith Kumar Nair, Prema Nedungadi, Aditya Kumar Sahu, Robin Kowalski, Sasangan Ramanathan, and Krishnashree Achuthan. Fake news research trends, linkages to generative artificial intelligence and sustainable development goals. *Heliyon*, 10(3), 2024.

Résume

Grâce à l'évolution rapide et à la popularité d'Internet, la diffusion de l'information est devenue beaucoup plus simple et immédiate. En plus de faciliter la vie, cette technologie a également contribué à la propagation de fausses informations, ce qui a eu des conséquences néfastes considérables pour les pays, les sociétés et les individus. Pour faire face à ce phénomène, de nombreuses études ont été menées pour détecter les fausses informations.

On a exploré différentes perspectives sur la détection automatique des fausses nouvelles, en utilisant différents modèles d'extraction de caractéristiques et de classification. Toutefois, les tests empiriques, les classifications et les comparaisons des techniques existantes demeurent encore restreintes.

Cette thèse réexamine les définitions et les perspectives des fausses nouvelles et propose une classification actualisée du domaine, basée sur divers critères.

En outre, nous réalisons une étude empirique approfondie afin d'évaluer différentes méthodes de représentation des caractéristiques et des méthodes de classification, en nous appuyant sur la précision et le coût de la computation. Selon nos expériences, il est démontré que les méthodes d'extraction des caractéristiques optimales diffèrent en fonction des particularités des jeux de données. La combinaison de couches CNN résiduelles multiscales et BiLSTM dans le modèle hybride proposé se révèle particulièrement performante pour détecter les dépendances locales et globales dans les données textuelles. Le modèle peut extraire des caractéristiques à divers niveaux de granularité grâce à l'architecture multiscale, tandis que la couche BiLSTM capture les dépendances à long terme et les informations contextuelles.

Mots-clefs (5) : Fausses nouvelles,Traitement de texte,Traitement du langage naturel,Algorithmes d'apprentissage,Architecture de réseau

Abstract

With the rapid evolution and popularity of the Internet, the dissemination of information has become much simpler and more immediate. In addition to making life easier, this technology has also contributed to the spread of false information, which has had significant negative consequences for countries, societies, and individuals. To address this phenomenon, numerous studies have been conducted to detect false information.

Various perspectives on automatic fake news detection have been explored, using different feature extraction and classification models. However, empirical testing, classification, and comparison of existing techniques remain limited.

This thesis reexamines the definitions and perspectives of fake news and proposes an updated classification of the field, based on various criteria.

Furthermore, we conduct an in-depth empirical study to evaluate different feature representation methods and classification techniques, focusing on accuracy and computational cost. Our experiments demonstrate that optimal feature extraction methods vary depending on the specifics of the datasets. The combination of multiscale residual CNN layers and BiLSTM in the proposed hybrid model proves particularly effective for detecting both local and global dependencies in textual data. The model can extract features at various levels of granularity due to its multiscale architecture, while the BiLSTM layer captures long-term dependencies and contextual information.

Keywords (5) : Fake news , Text processing Natural language processing, Learning algorithms; Network architecture.