REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA

Faculté des Mathématiques et de l'Informatique

Département d'Informatique

N° :……..

DOMAINE : Mathématiques et Informatique

FILIERE : Informatique

OPTION : …….

Mémoire présenté pour l'obtention
Du diplôme deMaster Académique

Par: Hamouda Hadjer

Djegham Fatima

Intitulé

Language Identification Using Bi-grams Technique, ML and DL Algorithms

Soutenudevant le jury composé de :

MOKHTARI Rabah	Université de M'sila	Président
GADRI Said	Université de M'sila	Rapporteur
CHALABI Baya	Université de M'sila	Examinateur

Année universitaire: 2022 /2023

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA

Faculté des Mathématiques et de l'Informatique

Département d'Informatique

N° :……..

DOMAINE : Mathématiques et Informatique

FILIERE : Informatique

OPTION : …….

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par: Hamouda Hadjer

Djegham Fatima

Intitulé

Identification de la Langue Basée sur la Technique de Bi-Grammes et les Algorithmes de ML et DL

Soutenudevant le jury composé de :

MOKHTARI Rabah	Université de M'sila	Président
GADRI Said	Université de M'sila	Rapporteur
CHALABI Baya	Université de M'sila	Examinateur

Année universitaire: 2022 / 2023

Dédicace:

Je dédie ce travail à

Mon père et ma mère, que Dieu les protège et prolonge leur vie

Mon mari est mon partenaire de vie

A mes professeurs et encadrants

À tous mes amis tout au long de ma carrière universitaire, en particulier mon collègue chercheur

A toute ma famille, en particulier ma soeur et mes proches proches et lointains.

Hadjer

Dédicace:

Je dédie ce mot à ma mère et mon père qui n'ont pas hésité à me soutenir et à encourager ma période d'études,car j'y ai trouvé une source de force et de confiance en moi,ainci qu'à mes chéres soeurs khawla et souad et à ses enfants Mimi et Toto et mon cher frère et oncle Al-saeed .....

et Arjalia pour votre soutien et pour une longue période de temps,et j'essaierai et une longue période d'investissement dans la période de ma vie jusqu'à ce que je rencontre dieu bonnes attentes et retour à vous de ce que vous m'avez donné.

Fatima Zohra

Remerciements:

Nous tenons particulièrement à remercier notre directeur de recherche**GADRI Said**pour son soutien continu, ses précieux conseils et sa patience tout au long de ce processus de recherche. Sa contribution a été essentielle à la réussite de ce travail. Nous remercions notre famille pour leur soutien et leurs encouragements, leur donnant de l'énergie et veillant à ce qu'ils ne s'effondrent pas sous la pression, Merci.

Table de matières

Liste des tableaux …… vi

List des Figure ……..…….. iix

Liste de abréviations……...xiv

INTRODUCTIONGENERAL……...1

Chapitre 1 : Identification de la Langue

1. Introduction.....5

2. Définition de l’Identification de la Langue.....6

3. Travaux connexes sur l’Identification des Langue.....6

3.1. Analyse de fréquences.....6

3.2. Analyse acoustique.....6

3.3. Modèles de langues.....6

3.4. Approches hybrides.....6

4. Domaines d’Utilisation de l’Identification de la Langue.....7

5.Les défis de l'identification de la langue.....7

5.1. La diversité linguistique.....7

5.2. Les influences culturelles.....7

5.3. Les erreurs de reconnaissance vocale.....7

5.4. Les défis technologiques.....7

5.5. La nature dynamique de la langue.....8

6. Caractéristiques de l’Identification de Langue.....8

6.1. Octets et codage.....8

6.2. Caractères.....8

7. Définition de Classification.....9

8. Objectifs de la classification.....10

9. Processus de classification.....10

9.1. Prétraitement (Preprocessing)....11
9.1.1. Ponctuation....11
9.1.2.Tokenization....11
9.1.3. Elimination des mots vides....11
9.2. Représentation de texte....12
9.2.1. Représentation en Sac de mot (bag of Word)....12
9.2.2. Representation Par des phrases :....13
9.2.3. Représentation avec des racines lexicales (tiges)....13
9.2.4.Représentation avec des lemmes....13
9.2.5.Représentation conceptuelle....13
9.2.6.Représentation basée sur les N-grammes....14
9.3.Méthodes de catégorisation des textes....14
9.3.1.Méthodes conventionnelles....14
9.3.2.Méthodes des plus proches voisins....15
9.3.3. Un modèle de codage vectoriel binaire....15
9.3.4. Méthode de n-grammes....15
9.4. Pondération de terme (Term Weighting/Term Coding)....16
9.4.1. Fréquence des termes TF (Term frequency)....16
9.4.2. Fréquence de documents inversés IDF (Inverse Document Frequency)....17
9.4.3.TF-IDF « Term Frequency Inverse Document Frequency»....17
9.4.4.Codage TFC....18
9.4.5.CodageLNU....18
10.Conclusion....19

CHAPITRE 2 :Apprentissage automatique et apprentissage profond

1.Introduction....21
2.Apprentissage automatique....21
2.1.Définition....21
2.2.Types d'apprentissage automatique....22
2.2.1. Apprentissage supervisé (Supervised learning)....22
2.2.2. Apprentissage non supervisé (Unsupervised learning)....22

2.2.3. Apprentissage par renforcement (Reinforcement learning).....22
2.2.4. Apprentissage semi-supervisé (Semi-supervised learning).....22
2.2.5. Apprentissage par transfert (Transfer learning).....22
3.Apprentissage profond.....24
3.1 Définition.....24
3.2 Réseau neuronal profond.....24
3.2.1 La fonction d'activation.....25
4. Types de réseaux de neurones profonds.....26
4.1. Réseaux de neurones convolutionnels (CNN).....26
4.2. Réseaux de neurones récurrents (RNN).....27
4.3. Longue mémoire à court terme.....27
4.4. Réseaux de neurones auto-encodeurs (AE) 28
4.5. Réseaux de neurones adverses génératifs (GAN) 28
4.6. Transformer 29
4.7. Réseaux de neurones prenant en compte les facteurs de temps (TCN).....29
5.Algorithmes deApprentissage……..…….29
6. Apport de ML et DL…….….30
7.Conclusion……..30

Chapitre 3:Implémentation et Expérimentation

1. Introduction.....32
2.L'environnement de développement.....32
2.1. Environnement matériel.....32
2.2. L'environnement logiciel.....32
2.3. Langage de programmation et Bibliothèque.....33
3.Data set utilisé.....36
4.Algorithmes utilisés.....37
4.1.Prétraitement.....37
4.1.1. Chargement des données.....37
4.1.2.Suppression de ponctuation.....38
4.1.3.Suppression des mots vides.....38
4.1.4. Modèle DNN.....38

4.2. Construire le formulaire.....39
4.3. Former le modèle.....39
5 .Évaluation du modèle.....40
6. Interface d'application.....41
7.Conclusion.....48
Conclusion générale…….. 50
Références.....51

Liste des tableaux

Table 01:	Extraits d'articles de Wikipédia sur la PNL dans différentes langues……..	**5**
Table 02:	Représentation binaire … ……	**15**
Table 03:	Représentation classique en n-gram du mot "corpus"……	**16**
Table04:	*tf* and *tf-idf* variants [18]……	**18**
Table05:	Exemple de Data set utilisé… ……	**38**
Table06:	le nombre de text pour chaque langue……	**39**
Table 07:	Description du modèle DNN proposé ……	**40**
Table 08:	Valeur de perte et valeur de précision obtenue ……	**42**

List des Figures

Figure1:	Processus de classification…….....	**10**
Figure2:	Paradigmes de l'intelligence artificielle…….....	**21**
Figure3:	Types d'apprentissage automatique…….....	**23**
Figure4:	D'apprentissage automatique… ……..	**24**
Figure 5:	Architecture de réseau neuronal profond …….....	**25**
Figure 6:	Exemple de Réseaux de neurones convolutionnels (CNN)……..…	**26**
Figure 7:	Exemple de Réseaux de neurones récurrents (RNN)……..	**27**
Figure 8:	Exemple de Longue mémoire à court terme…….	**27**
Figure 9:	Exemple de Réseaux de neurones auto-encodeurs (AE)……..	**28**
Figure 10:	Exemple de Réseaux de neurones adverses génératifs (GAN) …….	**28**
Figure 11:	Exemple de Transformer……..	**29**
Figure 12:	Exemple de Réseaux de neurones prenant en compte les facteurs de temps (TCN)……	**29**
Figure13:	Logo de spyder…….....	**30**
Figure14:	Logo de Anaconda…….....	**34**
Figure15:	logo de python…….....	**34**
Figure16:	la fonction de chargement des données…….....	**37**
Figure17:	de chargement des données dans spyder…….	**38**
Figure18:	la fonction de ponctuation…….	**38**
Figure19:	la fonction de suppression de mot vide…….....	**38**
Figure20:	A représenté le modèle entraîné …….....	**39**
Figure21:	Deux courbes représentent la précision de la formation par rapport à la précision de la validation et perte de formation versus perte de validation…….	**40**

Figure 22: Page de connexion pour l'un des algorithmes…….. **42**

Figure 23: Image affichée lors de la saisie de l'algorithme DNN……. **42**

Figure 24: Une image montrant comment ajouter un groupe de données…… **43**

Figure 25: Image montrant le début de l'apprentissage d'un ensemble de données par un algorithme Deep Neural Network……... **43**

Figure 26: Image montrant l'achèvement de la formation d'un ensemble de données par un algorithme Deep Neural Network…… **44**

Figure 27: Une image montrant l'écriture d'une phrase ou d'un mot en arabe et spécifiant sa langue…….. **44**

Figure 28: Une image montrant l'écriture d'une phrase ou d'un mot en anglais et spécifiant sa langue…… **45**

Figure 29: Une image montrant l'écriture d'une phrase ou d'un mot en française et spécifiant sa langue…. **45**

Figure 30: Image affichée lors de la saisie de l'algorithme KNN……. **46**

Figure 31: Une image montrant comment ajouter un groupe de données…… **46**

Figure 32: Image montrant le début de l'apprentissage d'un ensemble de données par un algorithme KNN **47**

Figure 33: Image montrant l'achèvement de la formation d'un ensemble de données par un algorithme KNN…… **47**

Figure 34: Une image montrant l'écriture d'une phrase ou d'un mot en arabe et spécifiant sa langue…….. **48**

Liste des abréviations

LI: Identification de la langue.

TF: Fréquence de terme.

IDF: Fréquence d'Inverse de document.

TF_IDF: Terme Fréquence - Fréquence Inverse du document.

SVM: Machine Vecteur de Support.

ML: Machine Learning.

DL: Deep Learning.

DNN: Deep Neural Network.

IA: Intelligence Artificielle.

RI: Recherche Information.

ML: Modélisation du Langue.

INTRODUCTION GENERAL

INTRODUCTION GENERALE :

L'intelligence artificielle (IA) est un domaine de l'informatique qui se concentre sur la création de machines et de programmes capables de simuler l'intelligence humaine et d'effectuer des tâches qui nécessitent normalement une intelligence humaine, telles que la reconnaissance de la parole, la compréhension du langage naturel, la prise de décision, la résolution de problèmes et l'apprentissage. L'IA implique l'utilisation de techniques de traitement du langage naturel, de vision par ordinateur, d'apprentissage automatique et de réseaux de neurones artificiels pour créer des systèmes intelligents qui peuvent s'adapter et s'améliorer au cours du temps. L'objectif de l'IA est de créer des machines qui peuvent penser et agir comme des êtres humains, mais avec des capacités améliorées telles que la vitesse, la précision et la capacité de traiter des quantités massives de données.

Les dernières années sont marquées par une augmentation énorme de la quantité d'information électronique rédigée en plusieurs langues. Dans le Traitement Automatique de la langue naturelle ceci devient une tache nécessaire et difficile à la fois. La détection de la langue est considérée comme une première étape importante pour tout une série de taches de traitement des langues [1].

Du fait que la détection de la langue intervient, de nos jours, dans diverses applications dans de nombreux domaines, on résulte qu’elle est devenue un sujet de recherche important. De ce fait, l'objectif de ce travail est de trouver une méthode automatisée basée sur les techniques de l’apprentissage automatique et l’apprentissage profond capable d'analyser des documents pour identifier la langue dans laquelle ils sont écris.

Le problème de la détermination de la langue d'un passage textuel écrit parmi un ensemble de langues potentielles est un problème complexe vu qu’en effet, un texte peut être écrit en plus d'une langue, mais pour des raisons de simplification, nous faisons ici l'hypothèse typique qu'un texte ne peut être écrit qu’en une seule langue et donc le système de détection de langue ne doit retourner qu’une seule langue correcte à la fin.

Et comme la plupart des recherches se concentrent sur les langues les plus parlées, alors que les langues peu dotées en ressources sont ignorées. Nous avons choisi une base de données variées entre des langues déjà traitées comme l’anglais et des langues en cours de traitement et difficile comme le persan et l'ourdou à cause du degré de richesse morphologique qu’est plus élevé dans ces langues.

La détection de la langue est très sensible à la langueur du texte et limitée par la quantité d'information disponible. Certains chercheurs supposent qu'un texte de langueur de plus de 300 caractères est plus facile à traiter qu'un texte de taille inférieure. Pour cette raison, nous avons essayé dans notre travail de maximiser les performances des algorithmes avec des segments de texte plutôt court jusqu'à un mot de trois caractères. La tache peut être modélisée comme un problème de classification et est basée sur l'apprentissage machine ou bien les approches linguistiques.

Présentation du domaine:

L'identification de la langue est un processus important pour de nombreuses applications linguistiques, telles que la traduction automatique, l'analyse des sentiments, la reconnaissance vocale, l'analyse du discours et bien d'autres.

En effet, les algorithmes de traitement de la langue et de l'apprentissage automatique nécessitent souvent de savoir dans quelle langue un texte ou un discours a été écrit ou prononcé. Ainsi, l'identification de la langue est la première étape pour de nombreuses tâches linguistiques, permettant de sélectionner un ensemble de règles de traitement de la langue appropriées pour une langue donnée.
Les méthodes d'identification de la langue peuvent être basées sur des différentes caractéristiques, telles que la morphologie, la syntaxe, la prononciation ou encore la fréquence des mots. Les algorithmes d'identification de la langue peuvent également combiner plusieurs caractéristiques pour améliorer les résultats.

Il existe plusieurs approches pour l'identification de la langue, l'approche morphologique qui est basée sur des règles, et utilise des listes de mots et des règles morphologiques pour identifier la langue, l'approche basée sur l'apprentissage automatique, qui utilisent des modèles préalablement entraînés pour identifier la langue.

L'identification de la langue est souvent un défi, en particulier pour les textes courts ou pour les textes écrits dans des langues proches. Cependant, grâce aux avancées de l'apprentissage automatique et des technologies de traitement de la langue, les méthodes d'identification de la langue continuent à s'améliorer et à jouer un rôle critique dans de nombreuses applications linguistiques.

Problématique du projet: La problématique de l'identification de la langue réside dans la difficulté de distinguer les différentes langues parlées dans le monde en se basant uniquement

sur des données textuelles ou orales. Les langues ont des similarités et des différences, et il y a des nuances dans les dialectes et les accents qui compliquent l'identification précise de la langue parlée. De plus, certains mots ou phrases peuvent être communs à plusieurs langues, rendant ainsi l'identification encore plus difficile. Cette problématique est particulièrement importante dans le traitement automatique du langage naturel, où il est crucial de savoir quelle langue est utilisée pour pouvoir fournir des réponses précises et personnalisées.

Notre travail consiste à développer un système qui identifie la langue d'un texte écrit en utilisant les algorithmes de l'apprentissage automatique et les techniques de deep learning.

Organisation du mémoire:

- **Une introduction générale:**Introduction générale à travers laquelle on a fait un survol sur l'intelligence artificielle et ses deux branches (ML, DL), ainsi que la problématique de notre projet (Language ldentification Using Bi-grams Technique, ML and DL Algorithms).

- **Un premier chapitre (Identification de la langue):**

  Dans lequel, nous présentons un aperçu sur la notion de l'Identification de la Langue LI, son historique, domaines d'application, ainsi que les défis et les caractéristiques de LI. Nous avons aussi présenté une brève définitionde la classification et son intérêt dans le domaine de l'Identification de la langue.

- **Un deuxième chapitre(Apprentissage automatique et apprentissage profond):**

  Ce chapitre présente brièvement l'apprentissage automatique et l'apprentissage profond en donnant: les concepts de base, les domaines d'application, l'évolution, avantages et inconvénients, et surtout les algorithmes et les modèles les plus utilisés.

- **Un troisième chapitre (Implémentation et Expérimentation):**

  Dans lequel nous présentons les résultats obtenus suite aux tests que nous avons menés pour résoudre le problème de Détection de la Langue, ainsi qu'une discussion des solutions apportées pour finir le chapitre avec l'exposition de l'application.

- **Une Conclusion générale:** Dans laquelle on a donné un résumé sur le travail qu'on a réalisé, les connaissances requises et les outils maitrisés lors de la réalisation du présent projet, les difficultés rencontrées, ainsi que les perspectives du projet.

Chapitre 1: Identification de la Langue

Chapitre 1 : Identification de la Langue

1. Introduction:

L'identification de la langue ("LI") est la tâche de déterminer la langue naturelle qu'un documentou une partie de celui-ci est écrit. Reconnaître un texte dans une langue spécifique vient naturellementà un lecteur humain familier avec la langue. Le tableau 1 présente des extraits de «Wikipedia articles » dans différentes langues sur le thème du traitement automatique du langage naturel (« NLP »), étiquetés.

Selon la langue dans laquelle ils sont écrits. Sans se référer aux étiquettes, les lecteurs de cet article reconnaîtra certainement au moins une langue dans le tableau 1 , et beaucoup sont susceptibles de pouvoir y identifier toutes les langues.

English	Natural language processing is a field of computer science, artificial intelligence and linguistics concerned with and human (natural) languages.
Italian	L'Elaborazione del linguaggio naturale `e il processo di trattamento automatico mediante un calcolatore elettronico delle informazioni scritte o parlate nel linguaggio umano o naturale.
Chinese	自然語言處理是人工智慧和語言學領域的分支學科。
Japanese	自然言語処理は、人間が日常的に使っている自然言語を コンピュータ に処理させる一連の技術であり、人工知能と言語学の一分野である。

Table 01: Extraits d'articles de Wikipédia sur la PNL dans différentes langues.

2. Définition de l’Identification de la Langue:

L'identification de la langue d'un texte dans un corpus de textes multilingues est une phase importante et critique le processus de catégorisation contextuelle des documents multilingues[2]. Cela peut être effectué manuellement par un locuteur ou un linguiste compétent, ou automatiquement à l'aide d'algorithmes informatiques. L'identification de la langue est utilisée dans un certain nombre de domaines, tels que la traduction, la reconnaissance vocale et la recherche d'informations en ligne.

3. Travaux connexes sur l’Identification des Langue:

En plus des technologies de reconnaissance vocale et de traitement du langage naturel, il existe d'autres travaux connexes sur l'identification des langues, tels que:

3.1. Analyse de fréquences: cette méthode consiste à analyser les fréquences de différents phonèmes dans un échantillon de parole pour identifier la langue. Chaque langue a une distribution de fréquences de phonèmes qui lui est propre, ce qui permet d'identifier la langue parlée.

3.2. Analyse acoustique: l'analyse acoustique consiste à étudier les caractéristiques acoustiques de la parole, telles que le timbre, la fréquence fondamentale, l'intensité, etc. Ces caractéristiques peuvent permettre d'identifier la langue parlée en comparant les résultats avec une base de donnéesde référence.

3.3. Modèles de langues: les modèles de langues sont des systèmes informatiques qui utilisent des statistiques pour identifier la langue parlée. Ils fonctionnent en analysant les motifs de séquences de phonèmes pour déduire la langue.

3.4. Approches hybrides: certaines approches combinent les méthodes ci-dessus pour obtenir des résultats plus fiables. Par exemple, une méthode hybride pourrait utiliser l'analyse de fréquences et l'analyse acoustique en combinaison avec un modèle de langue pour identifier la langue parlée.

Dans l'ensemble, l'identification des langues est un domaine en constante évolution, avec de nombreux développements récents dans les technologies de reconnaissance vocale et de traitement du langage naturel. De nouvelles méthodes sont continuellement étudiées pour améliorer la précision et la fiabilité de l'identification des langues, ce qui est essentiel pour de nombreuses applications, notamment la traduction automatique et l'analyse des sentiments dans différents langages.

4. Domaines d’Utilisation de l’Identification de la Langue:

1. Phonétique: étude des sons et de leur production dans une langue.

2. Morphologie: étude de la structure des mots et de leurs variations.

3. Syntaxe: étude de la structure des phrases et de leur organisation grammaticale.
4. Sémantique: étude du sens des mots et des phrases.

5. Pragmatique: étude de l'utilisation de la langue en contexte, et de son influence sur la communication.
6. Lexicologie: étude du vocabulaire d'une langue, de ses racines et de ses évolutions.

7. Sociolinguistique: étude des pratiques linguistiques dans leur contexte social, culturel, et historique.
8. Psycholinguistique: étude des processus cognitifs impliqués dans la compréhension et la production de la langue.

9. Neurolinguistique: étude des bases neurologiques de la langue et de la communication.
10. Langues et cultures de spécialité: études de l'environnement technique, économique ou étiqueté de certaines langues.

5.Les défis de l'identification de la langue:

5.1. La diversité linguistique: Le monde est rempli d'une grande diversité de langues, certaines ayant de nombreuses variantes et nuances. Cela rend difficile l'identification de la langue avec précision, surtout lorsqu'il s'agit de langues peu connues ou variantes régionales.

5.2. Les influences culturelles: Les dialectes et les accents sont souvent influencés par les cultures régionales et nationales, ce qui peut rendre l'identification de la langue difficile pour les non-locuteurs.

5.3. Les erreurs de reconnaissance vocale: Les programmes de reconnaissance vocale peuvent parfois mal identifier les langues, surtout si les participants parlent rapidement ou avec un accent étranger. Ces erreurs peuvent conduire à des résultats faussés.

5.4. Les défis technologiques: Les systèmes de reconnaissance vocale dépendent de la qualité de l'audio, de la qualité de la connexion Internet, de la fiabilité du logiciel et d'autres pièges technologiques qui peuvent affecter la reconnaissance des langues.

5.5. La nature dynamique de la langue: Les langues évoluent constamment, de nouvelles expressions et termes apparaissent, tandis que les expressions obsolètes disparaissent. Cela rend difficile l'identification précise de la langue utilisée dans la communication.

6. Caractéristiques de l'Identification de Langue:

Pour déterminer la langue dans laquelle un document d'entrée est rédigé, la décision est prise en fonction des caractéristiques du document, généralement au « niveau du mot » ou du « caractère ».Selon Tommi Jauhiainen et al. [3], la partie la plus importante d'un travail d'Identification de la Langue consiste à trouver des fonctionnalités qui peuvent être utilisées pour aider à la tâche. Certaines de ses caractéristiques sont résumées dans ce qui suit [3]:

6.1. Octets et codage:

Les documents sont numérisés en utilisant un codage particulier, qui transforme les caractères d'alphabet en séquence d'octets qui peuvent être stockée par la suite dans l'ordinateur, Certains codages sont spécifiques à une langue donnée (par exemple GuoBiao 18030 ou Big5 pour chinois et Shift-JIS pour le japonais), et d'autres comme la famille de codages Unicode sont spécifiquement conçus pour représenter le plus grand nombre de langues possible.

En 1996 Kikui [4] a introduit une étape de détection de l'encodage dans le traitement de LI mais vu qu'il était coûteux en termes de calcul, plusieurs chercheurs comme Mandl, Shramko,T artakovski &Womser-Hacker en 2006 [5]ont conclus à la possibilité d'ignorer l'encodage et partent du fait que tous les documents utilisent le même encodage par exemple toutes les données de Twitter et de Wikipédia sont codées en UTF-8.

6.2. Caractères:

¬ Caractères non alphabétiques ou non idéographiques

Simaki, Simakis, Paradis et Kerren en 2017 ont utilisé les fréquences de ponctuation etcertains symboles spéciaux comme caractéristiques pour distinguer les variantes anglaises[6].

¬Alphabets

En 1989 Henrich [7] a utilisé la connaissance des alphabets pour exclure les langues dans lesquelles un caractère propre à une langue n'apparaît pas dans un document de test.

¬Capitalisation

Parfois la capitalisation peut aider surtout le calcul des fréquences des caractères n-grammes mais en machine Learning quand on utilise l'orthographe d'un document on préfère la minuscule car il peut se produire une ambiguïté) par exemple : machine et MACHINE ne sont pas équivalents.

¬Le nombre de caractères dans les mots

Langer en 2001 [8] a été le premier qui a utilisé la longueur de mot pour l'identification de la langue, la méthode basée sur les mots est utilisée pour identifier les langues qui marquent les limites des mots comme par exemple le nombre moyen de lettres par mot en français varie entre 5 et 6 lettres par contre en allemand La taille du mot peut être égale 36.

¬ La fréquence de chaque caractère

Kerwin (2006) a utilisé les fréquences des caractères comme vecteurs caractéristiques. Par exemple la phrase : « L'identification de la langue d'un texte est utile dans un large éventail d'applications » contient 73 lettres dont 7 sont des "L", ce qui fait que la fréquence relative à cette lettre est de 0,10, soit 10 %. On se base ensuite sur ces fréquences pour détecter la langue. La figure 2 représente la fréquence relative de lettre « o » dans diverses langues [9].on peut aussi utiliser la probabilité des caractères.

7. Définition de Classification:

La classification de la langue est le processus de regrouper les langues en fonction de certaines caractéristiques linguistiques communes telles que la grammaire, le vocabulaire, la prononciation, l'histoire, la géographie, et la culture. La classification de la langue est utilisée pour aider à comprendre les relations entre les différentes langues et pour faciliter la communication entre les personnes qui parlent des langues différentes. Les systèmes de classification de la langue varient en fonction des perspectives et des objectifs de ceux qui les créent, et peuvent être basés sur des critères historiques, géographiques, linguistique ou culturels.

8. Objectifs de la classification:

a.Identifier la langue dans laquelle un texte est écrit.

b. Classer les langues en fonction de leur famille linguistique (indo-européenne, afro-asiatique, etc.).

c. Distinguer les langues selon leur époque (ancien français, moyen anglais, etc.).
d. Regrouper les langues en fonction de leur lieu géographique (langues africaines, langues

asiatiques, etc.).

e. Faire la distinction entre les langues officielles et les langues minoritaires.
f. Classer les langues selon leur statut sociolinguistique (langues maternelles, langues secondaires, etc.).

j. Repérer les langues en danger d'extinction.

9. Processus de classification:

Le processus de catégorisation est un système (Figure 3) qui reçoit en entrée un texte et en sortie associe une ou plusieurs catégories. Ceci est effectué en respectant un ensemble d'étapes. Ces étapes concernent le choix de collection des documents et le prétraitement, la représentation des textes, le choix de l'algorithme d'apprentissage, et l'évaluation desrésultats en vue de prévoir le degré de généralisation du classifier [10][11].

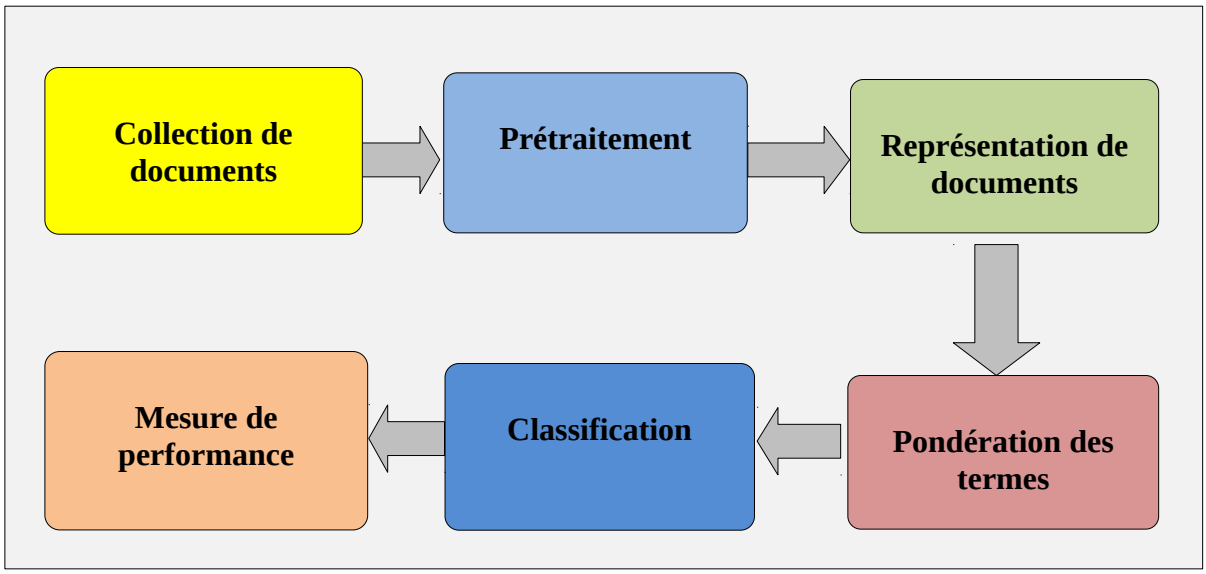

Figure 1: Processus de classification.

9.1. Prétraitement (Preprocessing):

La plupart des ensembles de données de textes et de documents contiennent de nombreuxmots inutiles tels que des mots vides, faute d'orthographe, etc. Dans de nombreuxalgorithmes, en particulier les algorithmes d'apprentissage statistiques et probabilistes,le bruit et les fonctionnalités inutiles peuvent avoir des effets néfastes sur lesperformances du système. Dans cette section, nous expliquons brièvement quelquestechniques et méthodes de nettoyage de texte et de prétraitement des ensembles dedonnées textuelles.

9.1.1. Ponctuation:

La ponctuation a pour but l'organisation de l'écrit grâce à un ensemble de signes graphiques comme (".",";", ":","!","?"), un seul signe de ponctuation peut modifier lanature d'une phrase,la ponctuation sert à faciliter la compréhension du texte, elle est un élément essentiel de la communication écrite, mais àcôté du traitement automatique de lalangue aucun algorithme peut comprendre ces signe.Donc convenu à enlever les caractèresspéciaux et les signes de ponctuations et des chiffres et etc....

9.1.2.Tokenization:

Un Token est une unité définie comme une séquence de caractères, La Tokenization estune méthode de prétraitement qui divise le texte, se produit à différents niveaux:

Untextepeut être divisé en paragraphes, phrases, mots, symboles ou phonèmes.

Exemple:

Après avoir dormi pendant quatre heures, il a décidé de dormir encore quatre heures/ Dans ce cas, le résultat de Tokenization par mot sera : *{“Après”, “avoir “, “dormir ”, “ pendant “,”quatre ”, “ heures ”, ”il ”, ”a ”, “décidé ”, “ de” , “dormir ” ,“ encore ”, “ quatre” ,“ heures ”}*.

9.1.3. Elimination des mots vides:

Une fois les documents textes découpés en Tokens, nous apercevons que certains de cesTokens sont présents dans tous les textes du corpus, c'est ce que nous appelons les motsvides (stopword en anglais) : les prépositions, les mots de liaisons, les déterminants,les adverbes, les adjectifs indéfinis, les conjonctions, les pronoms et les verbes auxiliairesetc... comme "la,le, dans, car, les " dans la langue française, et "the, and, to,by,after,ofdans la langue anglaise . (, بسيار

نام, اول, تواند, گرفتم,هايي , روي,) dans la langue person,dans la langue ourdou. Qui représente une grande part desmots d'un texte, mais malheureusement sont faiblement informatifs, sur le sens d'un texte puisqu'ils sont présents sur l'ensemble des textes. C'est mot ne contiennent pas aucuneinformation sémantique, qui ne modifie pas le sens des mots.

La classification des textes et des documents comprend bon nombre de ces mots qui necontiennent pas signification à utiliser dans les algorithmes de classification. La techniquela plus courante pour traiter ces mots est de les supprimer des textes et des documents[10][11]. Leur suppression réduit la taille du lexique du document, le temps de traitement et letemps d'apprentissage seront réduit considérablement.

9.2. Représentation de texte:

À chaque fois qu'il est question de définir un problème de façon à assurer un traitement automatique, il est impossible de passer l'étape où il faut choisir la façon dont on va représenter le problème. Dans le cas de la classification automatique de textes, les algorithmes d'apprentissage ne sont pas capables de traiter directement les textes, on doitopter pour une méthode efficace qui permet de représenter les instances à traiter, soit les textes.

Cette étape consiste généralement en la représentation de chaque document par un vecteur, dont les composantes sont par exemple les mots contenus dans le texte, afin de le rendre exploitable par les algorithmes d'apprentissage [12].

9.2.1. Représentation en Sac de mot (bag of Word):

Le modèle du sac de mots est une représentation réduite et simplifiée d'un texte, La technique de sac de mots est utilisée dans plusieurs domaines tels que la vision par ordinateur, TAL, le spam bayésien ainsi que la classification des documents et la recherche d'informations par apprentissage automatique. Cette représentation des textes exclut toute analyse grammaticale et toute notion de distance entre les mots. Un grand nombre d'auteurs comme Lewis en 1992 [13] utilisent les mots comme composantes du vecteur pour représenter les textes. Cette représentation consiste à transformer simplement les textes en vecteurs dont chaque composante représente un mot.

Exemple:

Document: "Soyer optimistes et tout ira pour le mieux".

Sac de mots:{"Soyer ", "optimistes ", "et ", " tout ","ira ", " pour ", "le ", "mieux"}.

9.2.2. Representation Par des phrases:

Un Certains nombres des chercheurs comme S. Marvin and S. Scott [14] proposent la phrase comme unité de représentation car parfois les phrases peuvent être plus informatives que les mots. Par exemple «traitement automatique de la langue» et «recherche informatique» car les phrases ont l'avantage deconserver l'informationrelative à la position du mot dans la phrase et possède un degré d'ambiguïté pluspetit.

9.2.3.Représentation avec des racines lexicales (tiges):

Dans la représentation Bag of Words, chacun dans ection (mot dérivé) est considéré comme un terme différent ; en particulier les différentes formes d'un verbe sont considérées comme différentes mots (croix, croix, barré) bien qu'ils soient en formes ectées du même verbe. Donc, cela peut augmenter la dimension de l'espace vectoriel représentant différents textes. Pour traiter ce problème, il est nécessaire de ne considérer que les racines des mots (tiges) que les mots entiers. Plusieurs algorithmes ont été proposés pour substituer les mots par leurs racines; le plus connu pour l'anglais est l'algorithme de Porter[15].

9.2.4.Représentation avec des lemmes:

La lemmatisation consiste à utiliser l'analyse grammaticale pour remplacer les verbes par leurs formes innitives et les noms par leurs formes au singulier. Le lemmetisation est donc plus compliquée à mettre en œuvre que la recherche de racines, car elle nécessite une analyse grammaticale des textes. Un algorithme efficace nommé “Tree-Tagger” a été développé pour de nombreuses langues : anglais, allemand et italien. Utilise les arbres de décision pour effectuer l'analyse grammaticale avec les de paramètres spéci ques à chaque langue[10, 16].

9.2.5.Représentation conceptuelle:

Est basé sur le formalisme vectoriel, mais les éléments du vecteur ne sont pas directement n'est plus associée à des termes d'indexation mais plutôt à des concepts. L'idée est de rassembler les mots synonymes et associer un concept lexical sous-adjacent qui nécessite la construction d'une base lexicale pour chaque langue (par exemple, ontologie Wordnet). Par exemple on associe aux synonymes (sommet, sommet, pic) le concept “peak”. L'avantage de la représentation conceptuelle est de réduire l'espace vectoriel de représentation en rassemblant les mots synonymes et en leur attribuant un concept commun. Contrairement à la représentation “Sac de Mots” qui associe à chaque mot une dimension dans le vecteur. Cependant,

l'inconvénient majeur de la représentation conceptuelle est l'absence de bases lexicales pour tous les langages qui permettent de telles représentations [11, 17].

9.2.6.Représentation basée sur les N-grammes:

Un N-gramme de X est défini comme une séquence de N X consécutifs. X peut être un caractère ou un mot [3]. Un N-gramme de caractère est donc une suite consécutive de N caractères. [8, 10, 11] qui ne peuvent pas être ordonnés (par exemple, les 3-grammes de la phrase “Hello Sir” sont: \Hel", \ell", \llo", \lo ", \o S", \ Si", \Sir" [5,13,18. Un document est constitué de la liste des N-grammes les plus fréquents dans l'ordre inverse de leurs fréquences. L'approche de la segmentation du texte en N-grammes caractéristiques présente plusieurs avantages, notamment:

- Tolérant aux fautes d'orthographe, de frappe et d'OCR.

- Cette approche est indépendante de la langue.

- Eviter l'utilisation de lemmatisation et de stemming sur le texte qui nécessite une ort algorithmique et linguistique.

- La segmentation en mots est difficile pour certaines langues, par exemple en arabe les noms et les sujets supplémentaires sont dans certains cas attachés aux verbes et les string est donc une phrase du type: (katabtouhou) (je l'ai écrit).

Dans notre travail nous avons utilisé les deux représentations: “bag of words” et “N-grams of personnages”[10, 11, 18].

9.3.Méthodes de catégorisation des textes:

9.3.1.Méthodes conventionnelles:

Plusieurs méthodes existent dans le domaine de la catégorisation de texte. Leur di culté communeest la très grande dimension. Parmi ces méthodes, on peut noter : les arbres de décision(ID3, C4.5, CART, . . .) [18], réseaux de neurones à rétropropagation, SVM etMéthodes RBF [9, 15].

9.3.2.Méthodes des plus proches voisins:

De nombreux algorithmes de catégorisation de texte sont basés sur le concept de distance (similarité). L'idée principale est de trouver le texte de l'ensemble d'apprentissage, qui est le plus proche dans distance au nouveau texte à classer, et d'attribuer sa catégorie au nouveau texte. Nous peut également augmenter le nombre k de textes les plus proches du nouveau texte si celui-ci est nécessaire. Dans ce cas, la catégorie du nouveau texte est la même que la majorité desleurs k plus proches voisins (la catégorie majoritaire). Le défi de ces méthodes est de savoir comment définir une métrique de similarité. Pratiquement, il existe plusieurs distances,les plus utilisées sont : La distance du produit scalaire (Inner product), la distance euclidienne,Distance cosinus, Manhattan, Dice, Jaccard et autres.

9.3.3. Un modèle de codage vectoriel binaire:

Certains algorithmes de machine Learning (ex : Naïve Bayes) nécessitent l'utilisation de valeurs binaires. Il sera alors nécessaire de définir une certaine valeur «seuil» d'occurrence, une cellule wij sera considérée comme vraie (1) ou fausse (0). Cequi correspond simplement à la présence ou l'absence d'un terme dans undocument [19].

Exemple :

D1 : le chien bleu a mangé un biscuit bleu

D2 : le chien bleu a mangé un biscuit rouge

	Chien	bleu	Mangé	biscuit	bleu	rouge
D1	1	1	1	1	1	0
D2	1	1	1	1	0	1

Table 02: Représentation binaire.

9.3.4. Méthode de n-grammes:

Dans la littérature, ce terme désigne quelque fois des séquences qui ne sont ni ordonnées ni consécutives ; par exemple un bigramme peut être composé de lapremière lettre et de la troisième lettre d'un mot Cavnar and Trenkle 1994 [11, 20], denombreux travaux ont montré l'efficacité des n-grammes (n variant de 1 à 5)comme méthode de représentation des textes pour leur catégorisation. L'avantageest qu'il n'est pas nécessaire de rassembler des connaissances linguistiques pourconstruire un classificateur. Les n-grammes sont également extrêmement simples à calculer pour un texte donné.Les n-grammes de longueur 1, 2 et 3 sont

généralement appelés respectivement uni-grammes, bi-grammes et trigrammes ; pour les longueurs n ≥ 4, on utilise leterme "N-gram" [10, 11]. Ils peuvent être consécutifs ou se chevaucher. Les bi-grammes de caractères consécutifs créés à partir de «corpus»la séquence de six caractères sont *co* et *rp*tandis que les bigrammes qui se chevauchent sont *co, or, et pr*... Les n-grammes qui se chevauchent sont le plus souvent utilisés dans la littérature.

Par exemple, la décomposition du mot "corpus" donne le résultat suivant :

N=1	**N=2**	**N=3**	**N=4**	**N=5**
_	_c	_co	_cor	_corp
c	co	cor	corp	corpu
o	or	orp	orpu	orpus
r	rp	rpu	rpus	rpus_
p	pu	Pus	Pus_	pus__
u	us	us_	us__	us___
s	s_	s__	s___	s____

Table 03: Représentation classique en n-gram du mot "corpus".

9.4. Ponderation de terme (Term Weighting/Term Coding):

L'étape de pondération ça aide à mesurer l'importance d'un terme dans un document, on peut calculer l'importance de terme à partir de considérations et interprétations statistiques. Pour calculer la ponderation, on distingue les méthodes suivantes dans le but de trouver lestermes qui représentent mieux le contenu de document.

9.4.1. Fréquence des termes TF (Term frequency):

On désigne par TF la fréquence d'un mot (descripteur) dans un texte donné. C'est un calculde fréquence très simple, mais qui s'avère efficace et pratique. TF est la fréquence de termedans un document qui prend en compte le nombre d'occurrences du terme dans le document.

$$TF\,t\,,d = \frac{nt\,,d}{N\,d} \textbf{(1)}$$

Où $n_{t,\,d}$ est la fréquence d'apparition du terme *t* dans le document *d* et $N_{\,d}$ est le nombre total des termes dans *d*.

Le principal inconvénient de la fréquence des termes est le fait qu'il est possible, et d'ailleurs c'est un cas très probable en pratique, qu'un terme apparait avec une fréquence assez grande dans tous les documents d'un corpus. Dans ce cas, le terme en question perd

toute sa notion de discrimination relative au degré de présence. Une autre notion vient rectifier ce cas exceptionnel, nommée IDF (Inverse Document Frequency).

9.4.2. Fréquence de documents inversés IDF (Inverse Document Frequency): Est une mesure de l'importance du terme dans l'ensemble du corpus, le nombre de documents dans lequel le terme apparaît qui prend en compte le nombre d'occurrence du terme dans le corpus, elle est calculable par :

$$IDF\,t = \log\left(\frac{|D|}{||d\,j, tj \in dj||}\right) (2)$$

Ou :

$|D|$: le nombre total de document dans corpus

$|\{dj : tj \in dj\}|$: Le nombre de documents où le terme apparait.

Si le terme est très présent dans tout le corpus, alors le rapport sera égal à 1 et IDF = 0 donc le terme est neutralisé.

9.4.3.TF-IDF « Term Frequency Inverse Document Frequency»:

Nous avons vu que la fréquence d'un terme dans un document joue un rôle important dans le calcul de son degré de discrimination. En revanche, la représentation d'un texte, dans le but de le classifier, ne dépend pas seulement de son contenu, mais elle est liée étroitement au corpus auquel le texte appartient, la rareté de ce terme au sein des autres documents du corpus s'avère aussi importante que sa fréquence (abondance) dans le document en question. Cette combinaison judicieuse de ces deux principes (abondance particulière et rareté générale) a engendré la pondération dite TF IDF.

Une technique d'optimisation couramment utilisée pour la catégorisation de textes est d'avoirdes représentations plus riches en informations pour calculer le poids de w_{ij}.

$$TF\text{-}IDF = TF_{t,d} \times IDF_t \textbf{ (3)}$$

Afin d'éviter les problèmes posés par les différentes longueurs de textes, on doit avoir recours à une représentation TF-IDF normalisée nomme le codage TFC et comme parexemple le codage LTC Buckley et al. En 1994 [10, 11, 20]qui tente de réduire les effets desdifférences de fréquences.

Term frequency tf	Document frequency df	Normalization
Natural value $tf_{t,\ d}$: number of times term t occurs in document d	$idf_t = Log\ (\frac{N}{df_t})$	None
Boolean value (binary model): $tf_{t,\ d} = \begin{cases} 1 & if\ f_{t,\ d} > 0 \\ 0 & otherwise \end{cases}$	1	1
Normalized with max term frequencies $tf_{t,\ d} = \frac{tf_{t,\ d}}{Max(tf_{t1,\ d}, tf_{t2,\ d}, \dots, tf_{\lvert V\rvert,\ d})}$	$idf_t = Log\ (\frac{N}{df_t})$	Cosine normalization $tf_{t,\ d} = \frac{tf_{t,\ d}}{\sqrt{(tf_{t1,\ d})^2 + (tf_{t2,\ d})^2 + \dots + (tf_{\lvert V\rvert,\ d})^2}}$
Normalized with sum of term frequencies $tf_{t,\ d} = \frac{tf_{t,\ d}}{\sum_{i=1}^{\lvert V\rvert} tf_{t_i,\ d}}$	$idf_t = Log\ (\frac{N}{df_t})$	
Normalized with logarithm value $tf_{t,\ d} = (1 + \text{Log}\ (tf_{t,\ d}))$	$idf_t = Max[0, Log\ (\frac{N - df_t}{df_t})]$	
Augmented value $tf_{t,\ d} = 0.5 + \frac{0.5 * tf_{t,\ d}}{Max(tf_{t,\ d})}$	$idf_t = Max[0, Log\ (\frac{N - df_t}{df_t})]$	

Table 4:*tf* and *tf-idf* variants[21].

9.4.4.Codage TFC :

Le cryptage TFC est similaire au cryptage tf-idf, mais il présente également un avantage Corriger les longueurs des textes avec normalisation cosinus, afin d'éviter de promouvoir les plus longsdocuments.

$$TFC(t_i, d_j) = \frac{tf - idf(t_i, d_j)}{\sqrt{\sum_{k=1}^{|v|} ¿¿¿¿}} \qquad (4)$$

D'autres codages sont utilisés tels que : le codage LTC [Buckley et al., 1994] qui tente de réduire lales effets des différences de fréquence, ou le codage basé sur l'entropie. [Dumais, 1991, Aas etEikvil, 1999, Jalam 2003], [21].

9.4.5.CodageLNU*:*

Les différents textes qui composent un corpus ont des tailles différentes. Donc c'estnécessaire d'en tenir compte lors du codage des termes. Selon Singhal [Singhal, 1996a, Nget al., 2000, Mathieu, 2000], il y a deux facteurs à considérer lorsque l'on travaille sur des textes longs :

- Les mots présents ont tendance à avoir des fréquences plus élevées.
- Les textes longs sont plus susceptibles de contenir des mots-clés différents.

Ainsi, pour tenir compte de ces deux facteurs, ils proposent le codage LNU, défini comme suit :

$$LNU = L * U \quad \textbf{(5.1)}$$

$$L = \frac{1+\log(TF(m,t))}{1+\log(\overline{TF}(m))} \quad \textbf{(5.2)}$$

$$L = \frac{1}{0{,}8+\frac{0{,}2*U(t)}{\overline{U}}} \quad \textbf{(5.3)}$$

TF(m,t):le nombre de mots dans le texte t.
$\overline{TF}(m)$:La fréquence moyenne dans le texte t.
U(t) : le nombre de termes uniques dans le texte t.

$\overline{U}$:Le nombre moyen de mots dans le groupe[21].

10.Conclusion:

Étudier les langages et les mécanismes nécessaires à la mise en œuvre et au traitement L'automatisation par les machines est un domaine d'étude en plein essor, parmi ce domaine Sélection de la langue. Dans le premier chapitre, nous nous sommes concentrés sur les concepts de base de la définition du langage qui est une tâche importante de traitement automatique du langage qui peut être modélisée comme un problème de classification basé sur l'apprentissage automatique. Dans ce chapitre nous avons présenté la classification automatisée, son processus, ses différentes étapes et applications de CT, nous avons également décrit les deux types de classification et les problèmes de classification les plus importants pour chaque langue, et dans le chapitre suivant nous présentons l'architecture de notre système, dont nous avons utilisé l'ensemble dans nos tests.

CHAPITRE 2 :

Apprentissage automatique et apprentissage profond

CHAPITRE 2:Apprentissage automatique et apprentissage profond

1.Introduction:

Dans ce chapitre, nous présentons les idées et concepts de base de l'apprentissage automatique et apprentissage en profond" et la différence entre eux, et introduisons la conception de notre projet avecla structure de la solution proposée, puis nous détaillons chaque étape solution proposée.

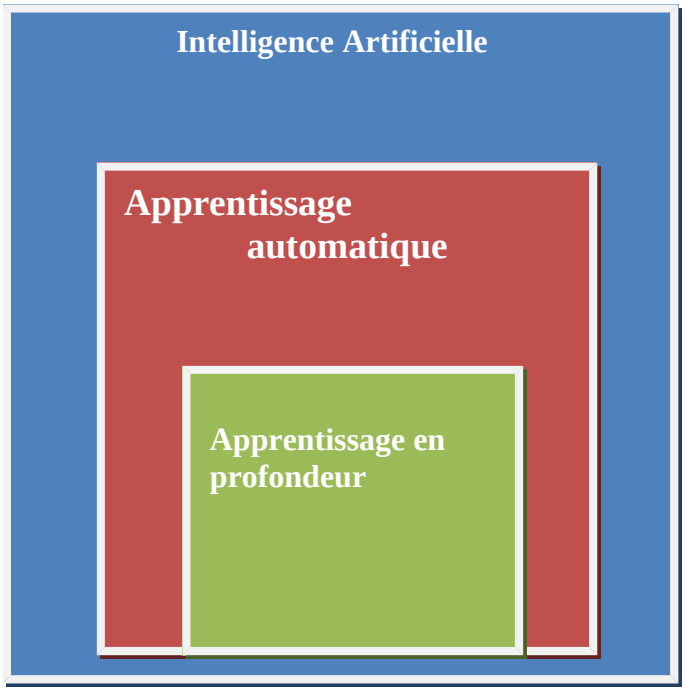

Figure2: Paradigmes de l'intelligence artificielle

2.Apprentissage automatique:

2.1.Définition:

L'apprentissage automatique simple (ou machine learning en anglais) est une branche de l'intelligence artificielle qui permet à une machine d'apprendre à partir de données, sans être explicitement programmée pour chaque tâche. Il s'agit d'un processus de découverte de relations inconnues dans les données grâce à des algorithmes statistiques et de modélisation mathématique. Les résultats de l'apprentissage machine peuvent être utilisés pour prendre des décisions, faire des prévisions ou classifier de nouvelles données. Les algorithmes

d'apprentissage simples incluent la régression linéaire, les arbres de décision et le k-plus proche voisin.

2.2.Types d'apprentissage automatique:

Il existe plusieurs types d'apprentissage automatique (machine learning) qui sont couramment utilisés. Voici quelques-uns des principaux types :

2.2.1. Apprentissage supervisé (Supervised learning): Dans ce type d'apprentissage, un modèle est entraîné à partir de données étiquetées, où chaque exemple de données est associé à une étiquette ou une valeur cible. Le modèle apprend à prédire la valeur cible pour de nouvelles données en se basant sur les exemples d'entraînement. Les algorithmes de régression et de classification font partie de l'apprentissage supervisé.

2.2.2. Apprentissage non supervisé (Unsupervised learning): Ici, les données d'entraînement ne sont pas étiquetées et le modèle cherche à découvrir des structures ou des modèles intrinsèques dans les données. L'objectif principal est de regrouper les données similaires ou de trouver des motifs intéressants. Les algorithmes de clustering et de réduction de dimensionnalité sont utilisés dans l'apprentissage non supervisé.

2.2.3. Apprentissage par renforcement (Reinforcement learning): Dans ce type d'apprentissage, un agent apprend à prendre des décisions séquentielles dans un environnement pour maximiser une récompense cumulative. L'agent interagit avec l'environnement et apprend par essais et erreurs, en ajustant ses actions en fonction des récompenses et des retours d'information reçus. L'apprentissage par renforcement est souvent utilisé pour les jeux, la robotique et les systèmes de recommandation.

2.2.4. Apprentissage semi-supervisé (Semi-supervised learning): Ce type d'apprentissage se situe entre l'apprentissage supervisé et non supervisé. Une partie des données d'entraînement est étiquetée, tandis que le reste est non étiqueté. L'objectif est d'utiliser à la fois les données étiquetées et non étiquetées pour améliorer les performances du modèle. L'apprentissage semi-supervisé peut-être utile lorsque l'étiquetage des données est coûteux ou difficile.

2.2.5. Apprentissage par transfert (Transfer learning): Dans cette approche, un modèle pré-entraîné sur une tâche source est utilisé comme point de départ pour une tâche cible similaire. Les connaissances acquises lors de l'entraînement sur la tâche source sont transférées et utilisées pour améliorer les performances sur la tâche cible. L'apprentissage par transfert est couramment utilisé pour les réseaux de neurones profonds.

Ces types d'apprentissage automatique peuvent être combinés ou adaptés en fonction des besoins spécifiques d'une tâche ou d'une application donnée.

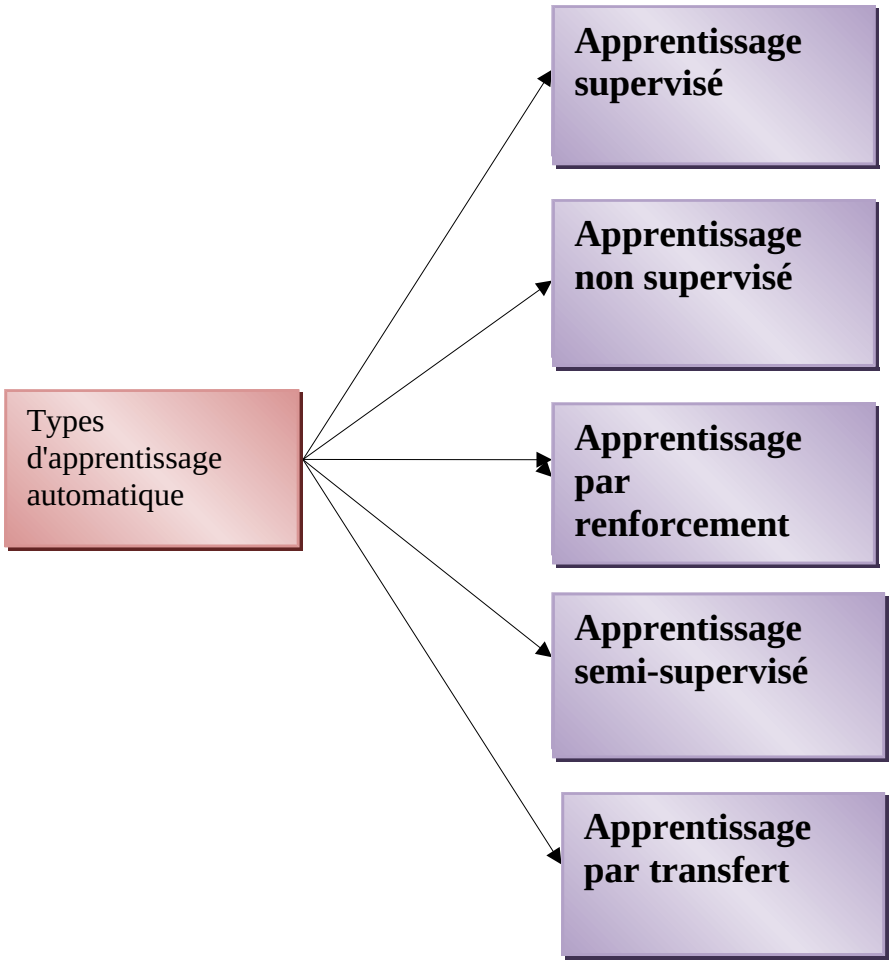

Figure 03: Types d'apprentissage automatique.

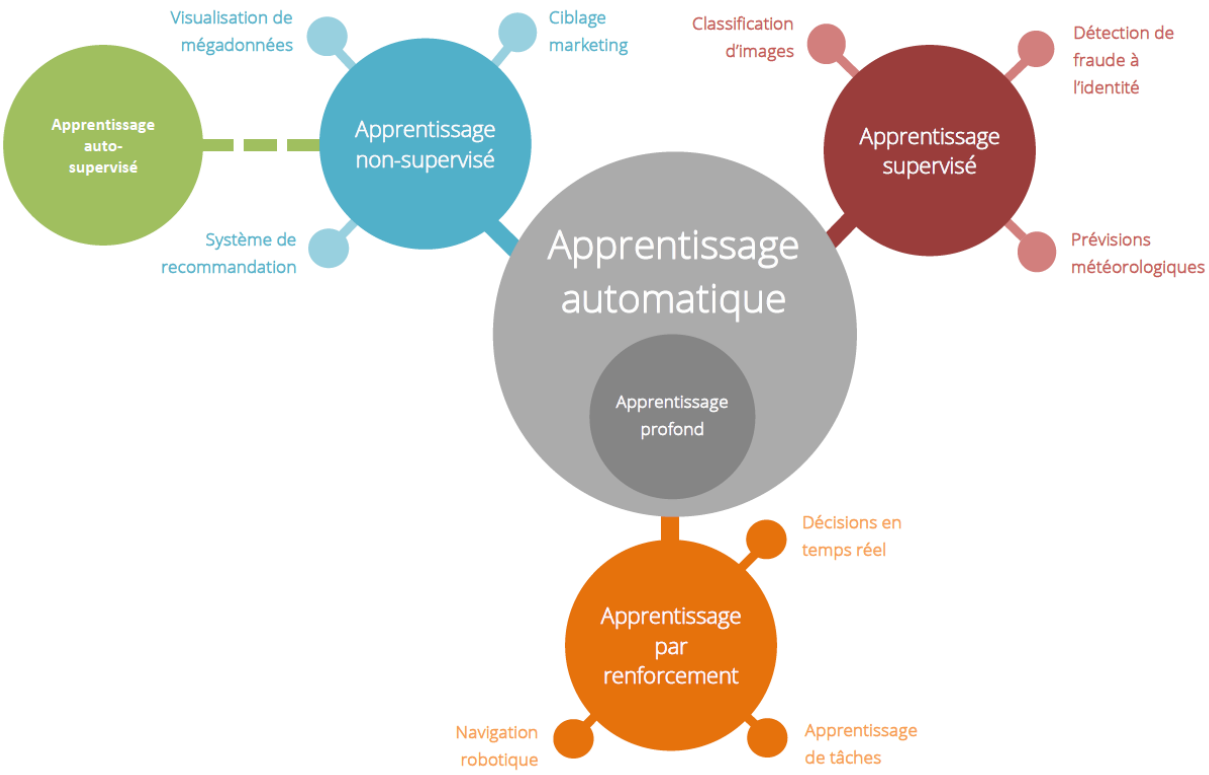

Figure 04: D'apprentissage automatique.

3.Apprentissage profond:

3.1 Définition:

L'apprentissage en profond simple est une branche de l'apprentissage automatique qui utilise des réseaux de neurones artificiels pour apprendre à partir de données non structurées. Il se compose d'une seule couche cachée de neurones et est souvent utilisé pour des tâches de classification ou de régression simples, telles que la reconnaissance d'images. C'est une méthode efficace pour traiter des données complexes et non linéaires.

3.2 Réseau neuronal profond:

DNN est l'acronyme de "Deep Neural Network", qui se traduit en français par "Réseau de Neurones Profond". Un DNN est un type spécifique de réseau neuronal artificiel qui possède plusieurs couches cachées entre la couche d'entrée et la couche de sortie.

Il est important de noter que le terme "DNN" est souvent utilisé de manière interchangeable avec "réseau de neurones profond" pour désigner des réseaux de neurones avec plusieurs couches cachées. Cependant, il convient de mentionner que d'autres types de réseaux de

neurones profonds, tels que les CNN (réseaux de neurones convolutionnels) et les RNN (réseaux de neurones récurrents), sont également couramment utilisés dans le domaine de l'apprentissage en profondeur.

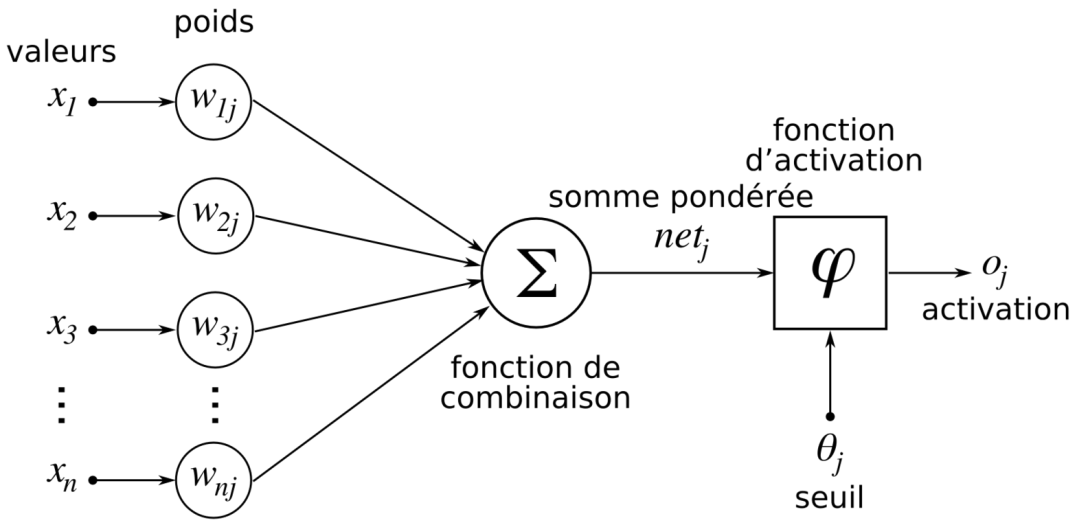

Figure 05: Architecture de réseau neuronal profond.

3.2.1La fonction d'activation:

La fonction d'activation d'un réseau neuronal profond est une fonction mathématique appliquée à chaque neurone pour déterminer l'activation de ce dernier. Elle est utilisée pour introduire de la non-linéarité dans le calcul effectué par le neurone. Les fonctions d'activation les plus courantes sont la fonction Sigmoïde, la fonction ReLU, la fonction Softmax, la fonction Tangente Hyperbolique, etc.

La fonction Sigmoïde est une fonction non linéaire qui transforme les valeurs d'entrée en une sortie comprise entre 0 et 1. Elle est souvent utilisée comme fonction d'activation pour la couche de sortie d'un réseau de neurones, car elle permet de produire des résultats de classification binaires.

La fonction ReLU (Rectified Linear Unit) est une fonction d'activation qui se comporte linéairement pour les valeurs positives et renvoie 0 pour les valeurs négatives. Elle est couramment utilisée dans les réseaux de neurones pour des problèmes de classification et de régression.

La fonction Softmax est également une fonction d'activation couramment utilisée dans les réseaux de neurones à plusieurs couches pour la classification de problèmes avec plusieurs classes. Elle convertit les valeurs d'entrée en une distribution de probabilités entre 0 et 1.

La fonction Tangente Hyperbolique ressemble à la fonction Sigmoïde, mais elle produit une sortie comprise entre -1 et 1. Elle est utilisée pour des problèmes de classification binaire et de régression.

En résumé, la fonction d'activation est une étape fondamentale dans le traitement de l'information dans un réseau neuronal profond, elle permet de donner une réponse non linéaire qui est essentielle à la résolution de problèmes complexes.

4.Types de réseaux de neurones profonds:

Il existe plusieurs types de réseaux de neurones profonds, notamment :

4.1.Réseaux de neurones convolutionnels (CNN): utilisés principalement dans la vision par ordinateur pour traiter des images et extraire des caractéristiques.

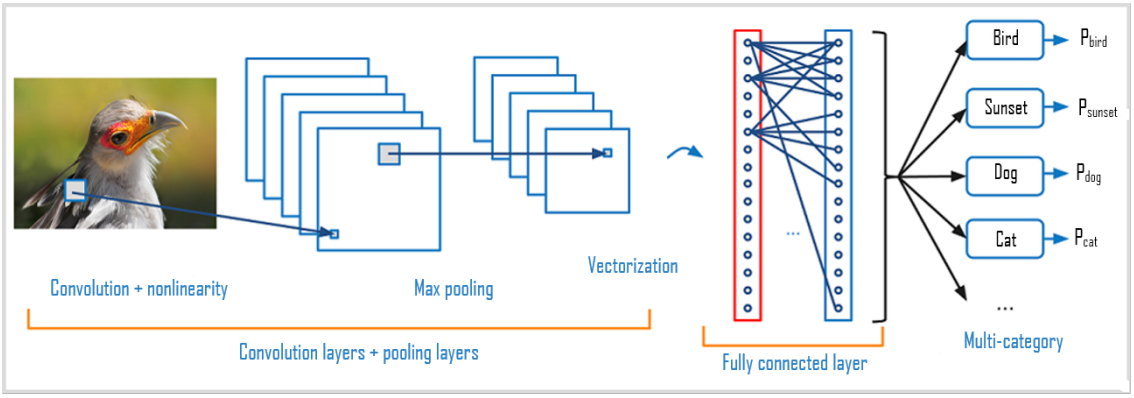

Figure 06: Exemple de Réseaux de neurones convolutionnels (CNN).

4.2.Réseaux de neurones récurrents (RNN): utilisés pour traiter des données temporelles telles que le langage naturel ou les séries temporelles.

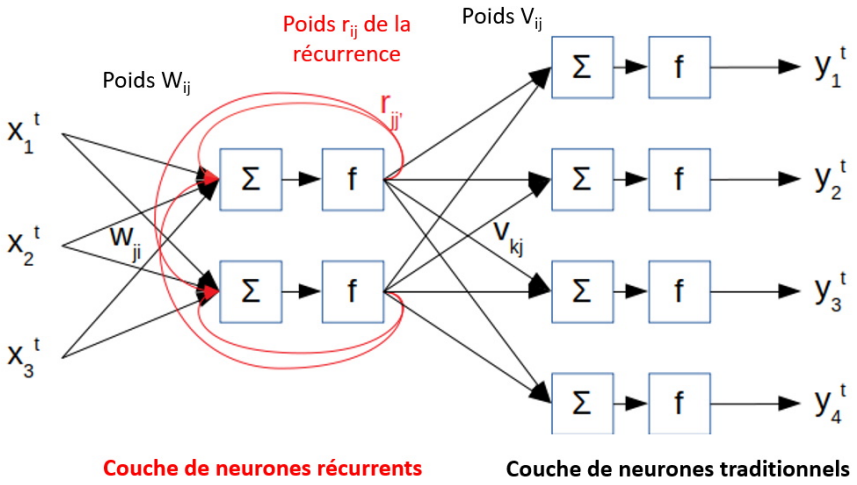

Figure 07: Exemple de Réseaux de neurones récurrents (RNN).

4.3.Longue mémoire à court terme: un type de RNN qui peut traiter des séquences plus longues et mémoriser des informations précédentes.

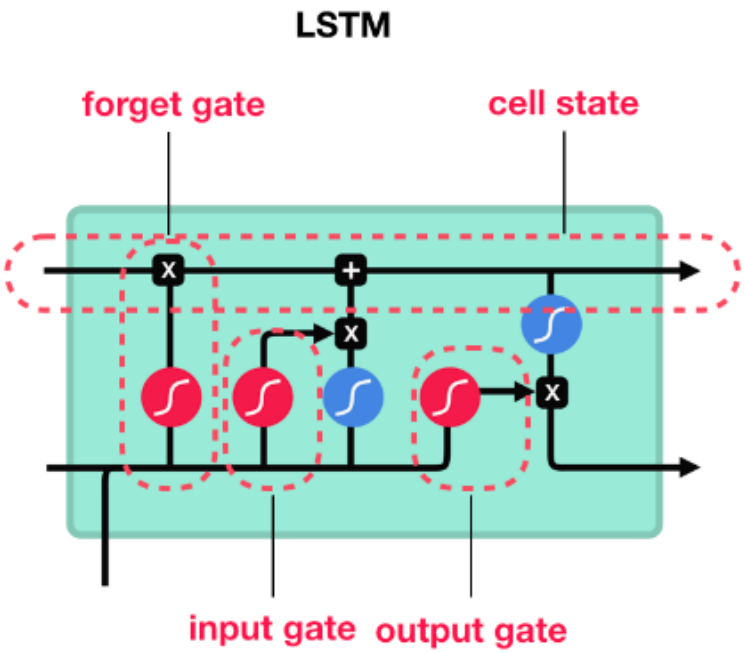

Figure 08: Exemple de Longue mémoire à court terme.

4.4.Réseaux de neurones auto-encodeurs (AE): utilisés pour la réduction de dimensionnalité, la reconstruction d'images ou la génération de données.

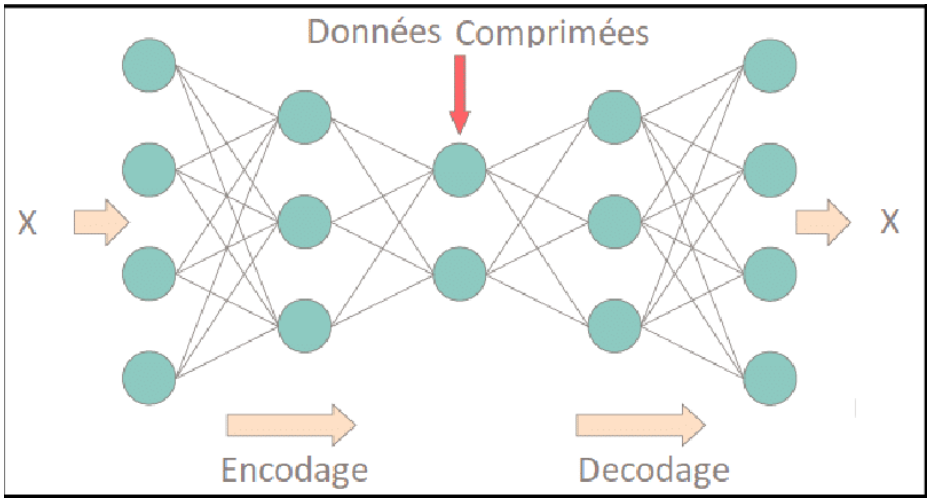

Figure 09: Exemple de Réseaux de neurones auto-encodeurs (AE).

4.5.Réseaux de neurones adverses génératifs (GAN): utilisés pour la génération de données synthétiques telles que des images ou du langage naturel.

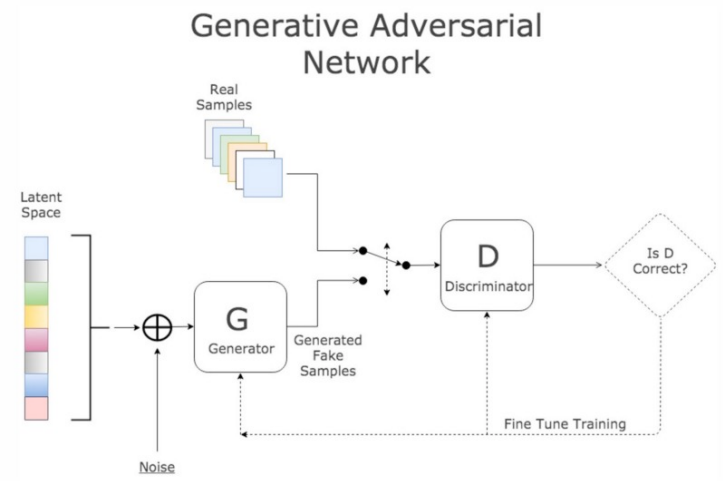

Figure10: Exemple deRéseaux de neurones adverses génératifs (GAN**).**

4.6.Transformer: un modèle d'apprentissage profond utilisé pour la traduction automatique, l'extraction d'informations et la génération de texte.

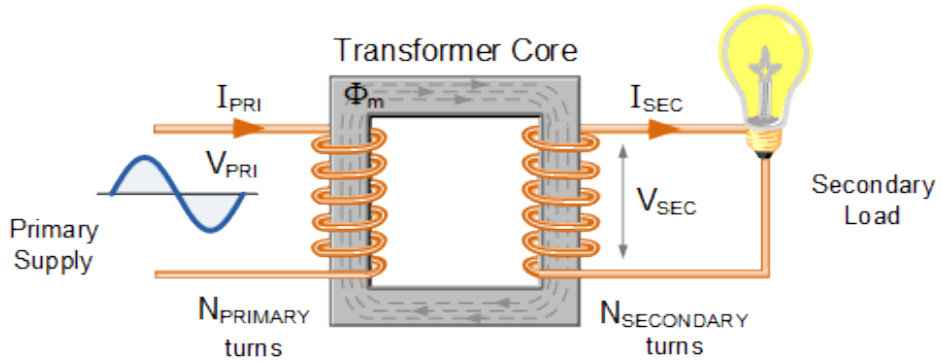

Figure11: Exemple de Transformer.

4.7.Réseaux de neurones prenant en compte les facteurs de temps (TCN): utilisés pour la prédiction de séquences et la reconnaissance vocale.

Ces réseaux de neurones peuvent être combinés pour former des architectures plus complexes, telles que les réseaux dits « hybrides » tels que le Convolutional Neural Network:Recursive Neural Networks (CNN-RNN).

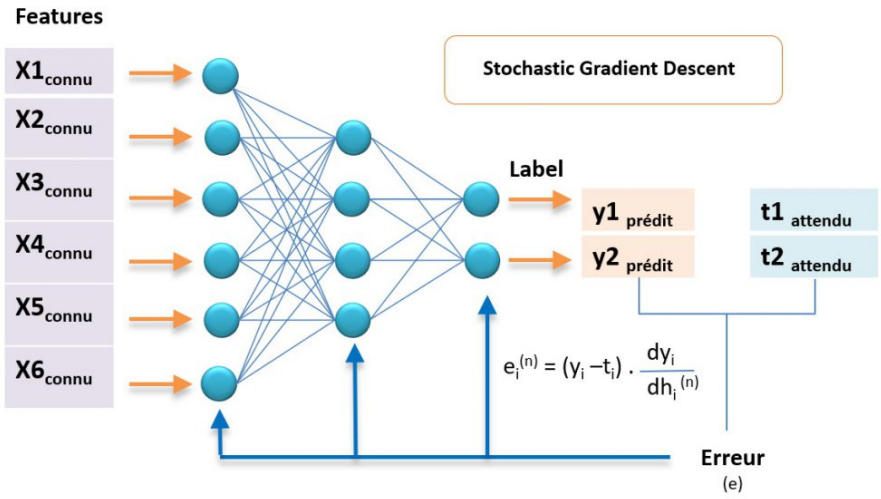

Figure12: Exemple de Réseaux de neurones prenant en compte les facteurs de temps (TCN).

5.Algorithmes de Apprentissage:

KNN:

L'algorithme des k-voisins les plus proches est une méthode d'apprentissage automatique qui consiste à classifier de nouveaux exemples en comparant leur éloignement à k exemples déjà classifiés. Pour ce faire, l'algorithme calcule la distance entre les attributs des exemples et utilise les k exemples les plus similaires pour prédire la classe de l'exemple à classer.

6. Apport de ML et DL

L’apprentissage automatique ML et l’apprentissage profound DL sont devenus les branches les plus populaires de l’IA pendant les dernières années. En particulier dans des domaines comme : la reconnaissance vocale, la vision par ordinateur, et autres domaines très intéressants. L’application des approches ML et DL a donné des résultats surprenants dans plusieurs domaines, notamment la vision par machine [22, 23], traitement de l’image [24, 26], détection des objets [27, 28], reconnaissance de chiffres et de lettres manuscrits [29, 30], optimisation des réseaux [31], réseaux de capteurs [32], analyse des sentiments [33, 34], détection de diabète [35]. L’approche DL donne des résultats meilleurs en terme exactitude de reconnaissance (accuracy), mais demande des quantités importantes de données et des équipements informatiques très sophistiqués. Dans notre projet, on a préféré d’appliquer les deux approches pour la détection de la langue d’un texte.

7.Conclusion:

L'apprentissage automatique et l'apprentissage en profondsont des domaines de l'informatique qui ont révolutionné la façon dont nous voyons les données et la manière dont nous les utilisons pour résoudre des problèmes. L'apprentissage automatique permet aux ordinateurs d'apprendre à partir de données, tandis que l'apprentissage en profond utilise des réseaux de neurones pour apprendre directement à partir des données. Ces technologies ont des applications dans de nombreux domaines, tels que la reconnaissance d'image, la reconnaissance vocale, la santé, la finance, etc. L'avenir de l'apprentissage automatique et de l'apprentissage en profond est très prometteur, avec des avancées constantes dans les domaines de l'IA et de l'apprentissage automatique, qui ont le potentiel de changer profondément notre manière de penser et de travailler. Cependant, il est également important de prendre en compte les enjeux éthiques et les limites potentielles de ces technologies, afin de s'assurer qu'elles sont utilisées de manière responsable et juste.

Chapitre 3:Implémentation et Expérimentation

Chapitre 3:Implémentation et Expérimentation.

1. Introduction:

Après avoir décrit notre solution de façon conceptuelle dans le chapitre 2.Nous exposonsdans ce chapitre la description de l'environnement de développement et les Langages et lesdifférents outils utilisés pour chaque étape de programmation. Ensuite nous montrons lesrésultats obtenus pour chaque partie.

2.L'environnement de développement:

Le choix de l'environnement de programmation convenable est très important pour le développement des projets. Cela se fait suivant plusieurs facteurs : la facilité d'utilisation, la disponibilité de plusieurs fonctionnalités et plusieurs bibliothèques, la communication avec d'autres environnements... etc.

2.1. Environnement matériel:

- Un ordinateur portable de type HP:

⎫ Système d'exploitation: Windows 10 professionnel 64 bits.

⎫ Processeur: Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz 2.40 GHz.

⎫Mémoire: 4GB.

- Un ordinateur portable de type HP:

⎫ Système d'exploitation: Windows 7 professionnel 64 bits.

⎫ Processeur: Intel® Core ™ i3-4005U CPU @1.70 GHz 1.70 GHz.

⎫ Mémoire: 4GB.

2.2. L'environnement logiciel

- **Spyder**

Spyder1 En 2008 Pierre Raybaut a créé et développé Spyder qui est un environnement scientifique puissant écrit en Python, pour Python. Spyder conçu par et pour des scientifiques, des ingénieurs et des analystes de données. Il présente une combinaisonunique des fonctionnalités avancées d'édition, d'analyse, de débogage et de profilage d'un outil de développement complet avec l'exploration de données, l'exécution interactive,

l'inspection approfondie et les superbes capacités de visualisation d'un package scientifique. En outre, Spyder offre une intégration intégrée avec de nombreux packages scientifiques populaires, notamment NumPy, Pandas, Scipy, etc. En comparaison avec d'autres IDE pour le développement scientifique, Spyder a un ensemble unique de fonctionnalités - multiplateforme, open-source, écrit en Python et disponible sous une licence non-copyleft.

Figure13: Logo de spyder.

2.3. Langage de programmation et Bibliothèques:

- **Anaconda:**

Anaconda est une distribution open source du langage de programmation Python couramment utilisé pour les tâches de science des données et d'apprentissage automatique. Il se compose d'un gestionnaire de packages, d'un gestionnaire d'environnement et de plusieurs autres outils pour faciliter le développement et le déploiement de workflows de science des données. Avec plus de 1500 packages de science des données inclus, Anaconda permet une installation et une gestion faciles des bibliothèques Python, ainsi que la création d'environnements de développement isolés pour différents projets. Anaconda propose également une interface utilisateur graphique (GUI) appelée Anaconda Navigator poursimplifier l'installation et la gestion des packages logiciels et des environnements.

Figure14: Logo de Anaconda.

- **Python:**

Python est un langage de programmation interprété et un environnement d'exécution, conçu pour la clarté de sa syntaxe et sa simplicité d'utilisation. Il est utilisé pour diverses tâches de programmation, telles que le développement d'applications bureautiques, de jeux, de sites web, d'outils de data science, de traitement du langage naturel et d'intelligence artificielle. Python est open source et multiplateforme, ce qui signifie qu'il peut être utilisé sur différents systèmes d'exploitation tels que Windows, Linux et MacOS.

Figure15: logo de python.

- **Bibliothèque RE:**

La bibliothèque RE (Regular Expression) est un outil de programmation qui permet de travailler avec des expressions régulières, qui sont des motifs de recherche et de manipulation de texte. Elle fournit des fonctionnalités pour rechercher, valider et manipuler des motifs spécifiques dans des chaînes de caractères. Grâce à cette bibliothèque, les développeurs peuvent effectuer des opérations avancées telles que la recherche de mots-clés, la vérification de formats de texte, le découpage de texte en fonction de règles spécifiques, et bien plus encore. Elle offre une solution puissante pour la manipulation de texte basée sur des motifs spécifiques.

- **Bibliothèque OS:**

Une bibliothèque OS est un ensemble de fonctions et de modules fournis par un système d'exploitation pour faciliter le développement d'applications en permettant aux développeurs d'interagir avec les fonctionnalités et les ressources du système d'exploitation. Elle agit comme une interface entre les programmes et le système d'exploitation, offrant des méthodes simplifiées pour effectuer des opérations telles que la gestion des fichiers, des répertoires, des processus, de la mémoire, du réseau, etc. Cela permet aux développeurs d'écrire des programmes plus efficaces, portables et fiables en utilisant des fonctionnalités préexistantes plutôt que de créer toutes les fonctionnalités à partir de zéro.

- **Bibliothèque Pandas:**

Pandas est une bibliothèque Python très populaire et puissante pour la manipulation et l'analyse de données tabulaires. Elle offre des structures de données flexibles appelées DataFrames, qui permettent de stocker et de manipuler facilement des données sous forme de tables. Pandas permet de charger des données à partir de différentes sources, de nettoyer et de transformer les données, de réaliser des opérations statistiques, de regrouper et d'agrégation des données, et de créer des visualisations. Grâce à ses fonctionnalités avancées, Pandas simplifie grandement les tâches d'analyse et de traitement des données, en faisant une bibliothèque essentielle pour les scientifiques des données et les analystes.

- **Bibliothèque Numpy:**

Numpy est une bibliothèque Python très populaire et puissante pour effectuer des calculs numériques et des opérations sur des tableaux de données. Elle fournit des structures de données efficaces appelées "ndarrays" qui permettent de stocker et de manipuler des données multidimensionnelles. Numpy offre des fonctionnalités avancées pour effectuer des opérations mathématiques, statistiques et d'algèbre linéaire sur ces tableaux. Grâce à sa rapidité d'exécution et à sa facilité d'utilisation, Numpy est devenue une bibliothèque incontournable pour les scientifiques des données, les chercheurs et les programmeurs qui travaillent avec des données numériques.

- **Bibliothèque Tensorflow:**

Une bibliothèque TensorFlow est un ensemble d'outils et de fonctions utilisé pour créer et entraîner des modèles d'apprentissage automatique. Elle permet de construire des graphes computationnels, où les opérations mathématiques sont représentées par des nœuds et les

données par des tenseurs. TensorFlow facilite la mise en œuvre d'algorithmes d'apprentissage en profondeur en fournissant des structures de données efficaces et des méthodes d'optimisation. Elle est utilisée dans de nombreux domaines tels que la vision par ordinateur, le traitement du langage naturel et les prévisions.

- **Bibliothèque Keras:**

Une bibliothèque Keras est une bibliothèque logicielle open-source largement utilisée pour créer et entraîner des réseaux de neurones artificiels. Elle fournit une interface conviviale et simplifiée pour construire des modèles d'apprentissage en profondeur, en utilisant des couches prédéfinies et des outils d'optimisation. Keras facilite le processus de développement et de déploiement de modèles d'apprentissage automatique, en offrant une abstraction de haut niveau et en prenant en charge plusieurs frameworks sous-jacents, tels que TensorFlow.

3. Dataset utilisé:

Un data set est un ensemble de données qui a été collecté et organisé en vue d'une analyse ultérieure. Il peut contenir des informations structurées ou non structurées, et peut être utilisé dans divers domaines tels que la recherche scientifique, l'analyse de marché, la modélisation statistique, etc. Le choix d'un data set dépend des besoins spécifiques de l'utilisateur et des variables clés qu'il souhaite étudier,Comme notre jeu de données contient 10 339 textes et 17 langues, ils sont (English, French, Spanish, Portugeese, Italian, Russian, Sweedish, Malayalam, Dutch, Arabic, Turkish, German, Tamil, Danish, Kannada, Greek, Hindi).

Text	Text	Langue
0	Nature, in the broadest sense, is the natural, physical, material world or universe…..	English
1	"Nature" can refer to the phenomena of the physical world, and also to life in general….	English
2	The study of nature is a large, if not the only, part of science….	English
3	Although humans are part of nature, human activity is often understood as a separate ….	English
4	The word nature is borrowed from the Old French nature and is derived from the Latin ..	English
……	……	……
10332	ನಿಮ್ಮತಪ್ಪುಏನುಬಂದಿದೆಯೆಂದರೆಆದಿನದಿಂದನಿಮಗೆಒ...	Kannada
10333	ನಾರ್ಸಿಸಾತಾನುಮೊದಲಿಗೆಹೆಣಗಾಡುತ್ತಿದ್ದಮಾರ್ಗಗಳನ್...	Kannada
10334	ಹೇಗೆ' ನಾರ್ಸಿಸಿಸಮ್ಈಗಮರಿಯನ್ಅವರಿಗೆಸಂಭವಿಸಿದಎ...	Kannada
10335	ಅವಳುಈಗಹೆಚ್ಚುಚಿನ್ನದಬ್ರೆಡ್ಬಯಸುವುದಿಲ್ಲಎಂದು ...	Kannada
10336	ಟೆರ್ರಿನೀವುನಿಜವಾಗಿಯೂಆದೇವದೂತನಂತೆಸ್ವಲ್ಪಕಾಣು...	Kannada

Table 05: Exemple de Data set utilisé.

Langue	Nombre de textes
English	1385
French	1014
Spanish	1014
Portugeese	739
Italian	698
Russian	692
Sweedish	676
Malayalam	594
Dutch	546
Arabic	536
Turkish	474
German	470
Tamil	469
Danish	428
Kannada	369
Greek	365
Hindi	63

Table 06: le nombre de textes pour chaque langue.

4.Algorithmes utilisés:

4.1.Prétraitement:

4.1.1.Chargement des données :

Cette instruction lit le fichier CSV situé dans le chemin spécifié et le charge dans un DataFrame appelé data.

```
data=pd.read_csv("C:/Users/Lahmar Info/Desktop/LangDetec/Lang_Detect.csv")


print(data.shape)
print(data)
```

Figure16: la fonction dechargement des données.

```
(10337, 2)
                                                    Text Language
0       Nature, in the broadest sense, is the natural...  English
1      "Nature" can refer to the phenomena of the phy...  English
2      The study of nature is a large, if not the onl...  English
3      Although humans are part of nature, human acti...  English
4      [1] The word nature is borrowed from the Old F...  English
...                                                  ...      ...
10332  ನಿಮ್ಮ ತಪ್ಪು ಏನು ಬಂದಿದೆಯೆಂದರೆ ಆ ದಿನದಿಂದ ನಿಮಗೆ ಒ...  Kannada
10333  ನಾರ್ಸಿಸಾ ತಾನು ಮೊದಲಿಗೆ ಹೆಣಗಾಡುತ್ತಿದ್ದ ಮಾರ್ಗಗಳನ್...  Kannada
10334  ಹೇಗೆ ' ನಾರ್ಸಿಸಿಸಮ್ ಈಗ ಮರಿಯನ್ ಅವರಿಗೆ ಸಂಭವಿಸಿದ ಎ...  Kannada
10335  ಆವಳು ಈಗ ಹೆಚ್ಚು ಚಿನ್ನದ ಬ್ರೆಡ್ ಬಯಸುವುದಿಲ್ಲ, ಎಂದು ...  Kannada
10336  ಟೆರ್ರಿ ನೀವು ನಿಜವಾಗಿಯೂ ಆ ದೇವದೂತನಂತೆ ಸ್ವಲ್ಪ ಕಾಣು...  Kannada
```

Figure17: dechargement des données dans spyder.

4.1.2.Suppression de la ponctuation:

```
    # removing the symbols and numbers
text = re.sub(r'[!@#$(),n"%^*?:;~`0-9]', ' ', text)
```

Figure18: la fonction de ponctuation.

4.1.3.Suppression des mots vides:

```
text = re.sub(r'[[]]', ' ', text)
```

Figure19: la fonction de suppression de mot vide.

4.1.4. Modèle DNN:

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 256)	1280256
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 128)	32896
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 64)	8256
dropout_2 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 32)	2080
dropout_3 (Dropout)	(None, 32)	0
dense_4 (Dense)	(None, 17)	561

Table 07: Description du modèle DNN proposé.

4.3. Former le modèle:

```
model.fit(X, Y, epochs=15, batch_size=32, validation_split=0.1)
#model.fit(x_train, y_train, epochs=15, batch_size=32, validation_split=0.1)
```

Figure20:A représenté le modèle entraîné.

Modèle DNN		
Epoch=15	**val_loss**	**val_accuracy**
1	*2.2488*	*2.64E-01*
2	*1.2484*	*0.6152*
3	*0.808*	*0.7673*
4	*0.5354*	*0.835*
5	*0.4797*	*0.8655*
6	*0.4043*	*0.8873*
7	*0.3618*	*0.8965*
8	*0.3377*	*0.9023*
9	*0.3332*	*0.9098*
10	*0.3048*	*0.9152*
11	*0.2768*	*0.9211*
12	*0.2546*	*0.9253*
13	*0.2539*	*0.9251*
14	*0.2435*	*0.9288*
15	*0.2494*	*0.9241*
	Train loss	Train Accuracy
	1.3675271272659302	**0.89861661195755**

Table 08: Valeur de perte et valeur de précision obtenue.

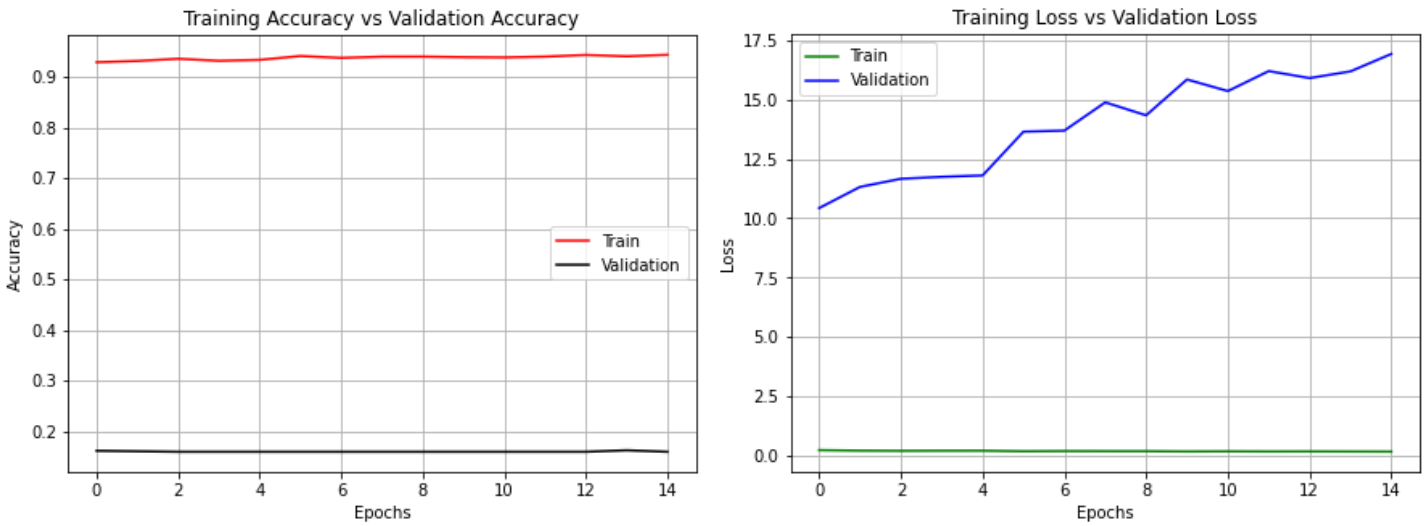

Figure21: Deux courbes représentent la précision de la formation par rapport à la précision de la validation et perte de formation versus perte de validation.

5 .Évaluation du modèle:

- **Accuracy:**

L'accuracy (ou précision) est un terme utilisé pour évaluer les performances d'un modèle dans le domaine de l'apprentissage automatique. En français, on l'appelle également "exactitude".

L'accuracy mesure la capacité d'un modèle à prédire correctement les classes des échantillons. Elle est calculée en divisant le nombre d'échantillons correctement classés par le nombre total d'échantillons.

Par exemple, si un modèle de classification est testé sur 100 échantillons et qu'il prédit correctement 90 d'entre eux, alors l'accuracy serait de 90%. Cela signifie que le modèle a une précision de 90% dans ses prédictions.

- **Loss:**

La "loss" (ou perte) est une métrique utilisée pour évaluer l'erreur ou la différence entre les prédictions d'un modèle et les valeurs réelles des données.

La loss mesure à quel point les prédictions du modèle s'écartent des valeurs réelles. L'objectif principal lors de l'entraînement d'un modèle est de minimiser cette perte afin d'améliorer ses performances.

- **Optimizer:**

L'optimizer (optimiseur) est un composant essentiel pour ajuster les paramètres d'un modèle d'apprentissage automatique afin de minimiser la loss (perte).

- **Batch size:**

Le "batch size" (taille du lot) correspond au nombre d'échantillons d'apprentissage traités simultanément par le modèle lors de l'entraînement.

- **Epochs:**

"Epochs" (époques) représente le nombre de fois que l'ensemble des données d'apprentissage est passé à travers un modèle lors de l'entraînement.

6. Interface d'application:

Notre interface se compose de trois pages:

6 .1.La première page:contientune adresse et un login pour l'un des algorithmes.

6.2. La deuxième page:

L'algorithme KNN:

Ajouter un ensemble de données puis l'étudier à l'aide d'un algorithme KNN et extraire l'exactitude de l'algorithme 0,70. Également sur cette page, il y a un endroit pour écrire une phrase ou un mot qui nous montre sa langue.

6.3. La troisième page:

L'algorithme DNN:

Ajouter un ensemble de données puis l'étudier à l'aide d'un algorithme DNN et extraire l'exactitude de l'algorithme 0,95. Également sur cette page, il y a un endroit pour écrire une phrase ou un mot qui nous montre sa langue.

Observation:

On remarque que l'algorithmeDNN a atteint une plus grande précision qu'un algorithmeKNN.

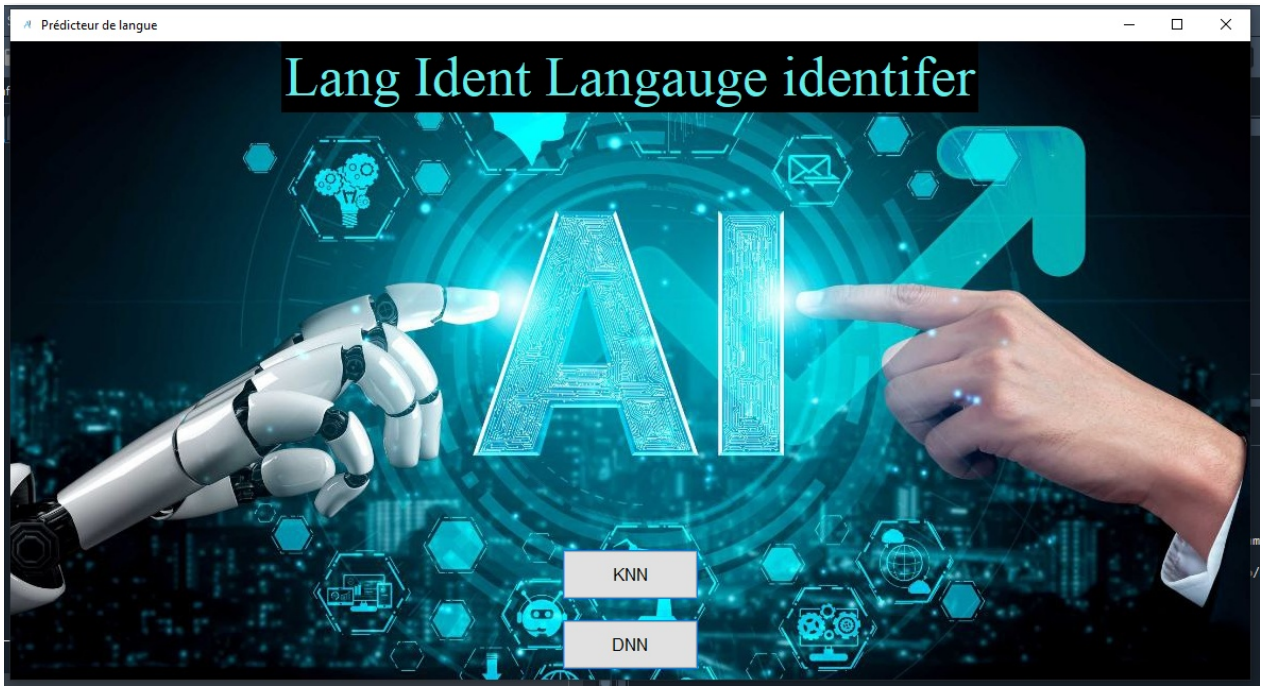

Figure22: Page de connexion pour l'un des algorithmes.

Figure23: Image affichée de l'algorithme DNN.

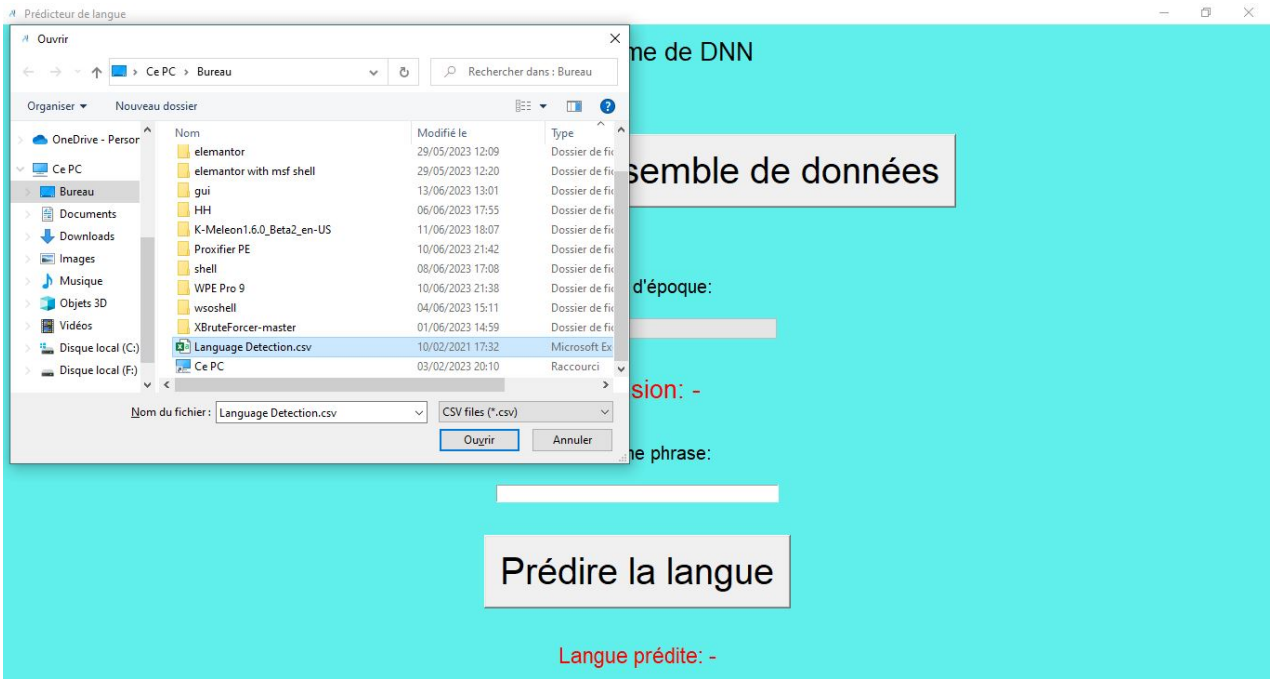

Figure24: Une image montrant comment ajouter un dataset.

Figure25: Image montrant le début de l'apprentissage d'un ensemble de données par un algorithmeDeep Neural Network.

Figure26: Image montrant l'achèvement de la l'apprentissage d'un ensemble de données par un modèleDeep Neural Network.

Figure27: Une image montrant l'écriture d'une phrase ou d'un mot en arabe et spécifiant sa langue.

Figure28: Une image montrant l'écriture d'une phrase ou d'un mot en anglais et spécifiant sa langue.

Figure29: Une image montrant l'écriture d'une phrase ou d'un mot en françaiseet spécifiant sa langue.

Figure30: Image affichée lors du lancement de l'algorithme KNN.

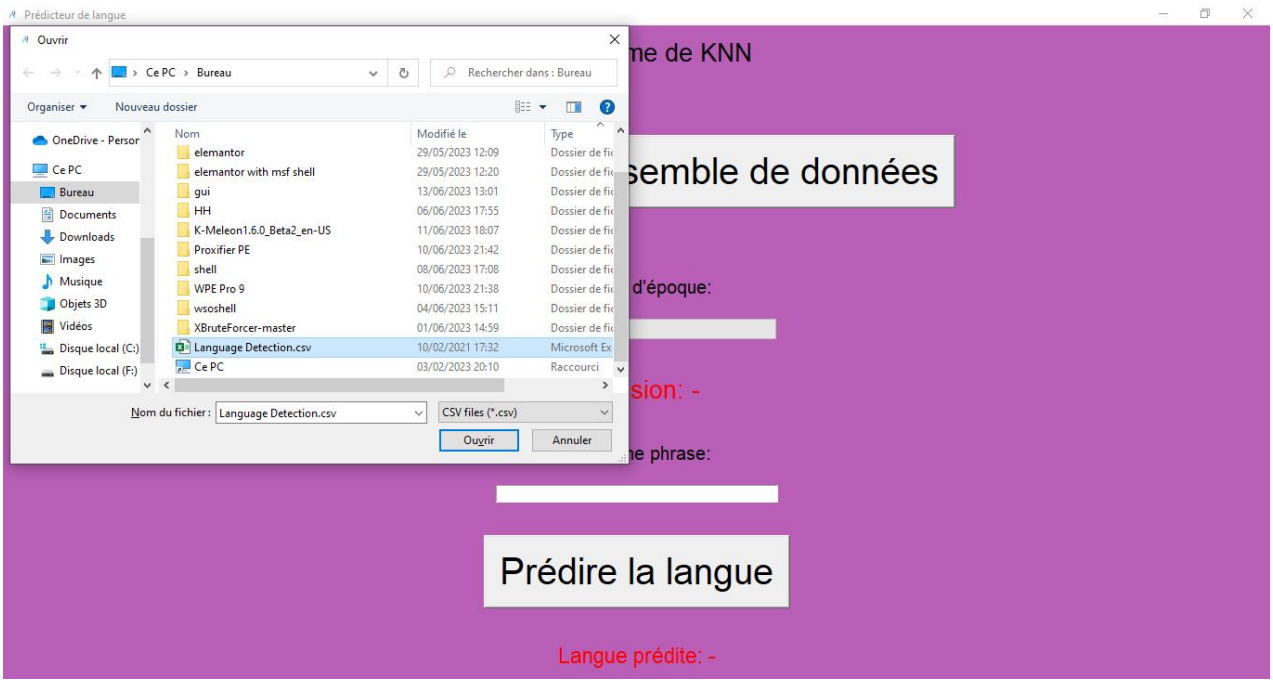

Figure31: Une image montrant comment ajouter un dataset.

Figure32: Image montrant le début de l'apprentissage d'un ensemble de données par un algorithmeKNN.

Figure33: Image montrant l'achèvement de l'apprentissage d'un ensemble de données par un algorithmeKNN.

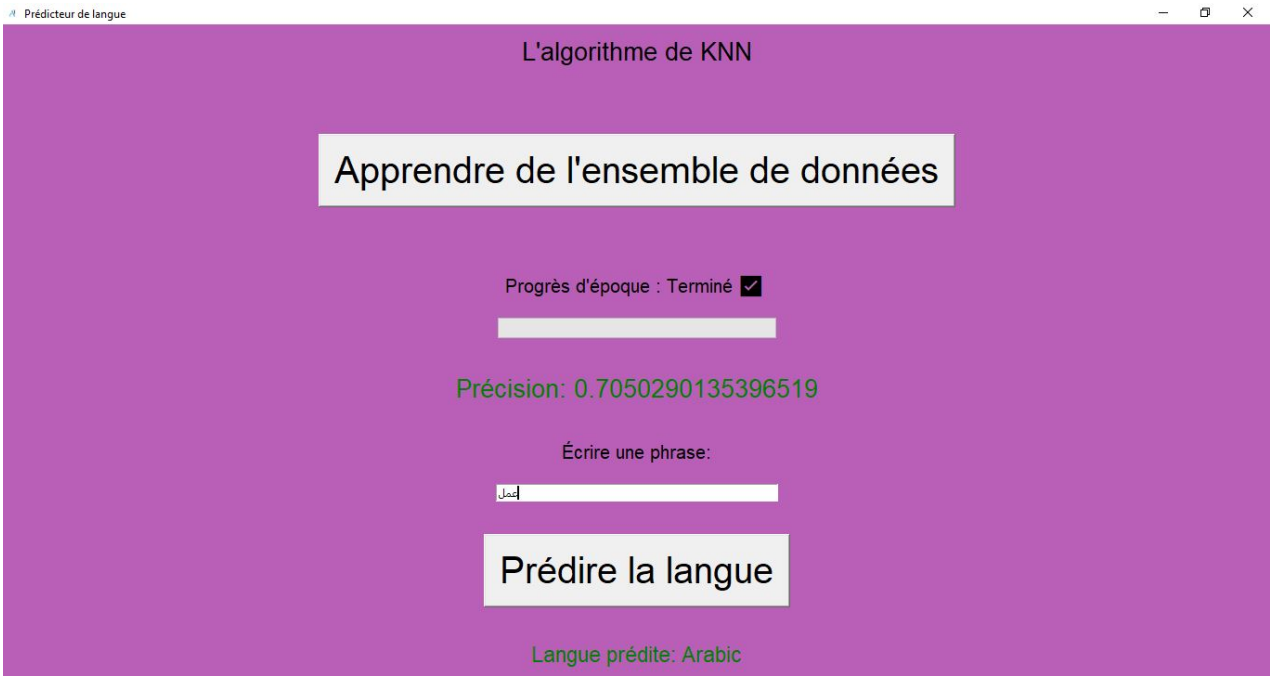

Figure34: Une image montrant l'écriture d'une phrase ou d'un mot en arabe et spécifiant sa langue.

7.Conclusion:

Au cours du présent chapitre, nous avons effectué les taches suivantes :

- Présentation des modèles DNN et KNN qui ont été implémentés.
- La mise en œuvre d'une interface utilisateur.
  - Nous avons également défini le modèle proposé, expliqué les résultats obtenus, suggéréquelques idées de travaux futurs, présenté les concepts de base et notre interface d'application.

Conclusion générale

Conclusiongénérale:

L'intelligence artificielle (IA) est de plus en plus intégrée dans notre vie quotidienne, notamment dans le domaine de l'identification de la langue. Grâceà l'IA, les machines sont désormais capables de comprendre et de traduire différents langages, ce qui facilite la communication entre les personnes de différentes cultures et nationalités. Cependant, l'IA n'est pas encore parfaite et peut encore faire des erreurs dans l'identification de la langue.

Les variations dialectales et les nuances culturelles peuvent être difficiles à reconnaitre, ce qui peut entrainer des erreurs d'identification. Il est donc important que les utilisateurs de ces technologies sachent que l'IA n'est qu'un outil et qu'il est toujours conseillé de vérifier les résultats obtenus. Il est important de noter également que l'IA ne doit pas être considérée comme un remplacement total de l'expertise humaine dans l'identification de la langue.

La compréhension du contexte culturel d'une langue est souvent essentielle pour une identification précise, et cela ne peut être fourni que par des experts humains.

En fin de compte, l'IA représente une avancée importante dans l'identification de la langue, mais elle doit être utilisée avec précaution et en complément de l'expertise humaine.

En travaillant ensemble, les humains et les machines peuvent obtenir les meilleurs résultats possibles dans le domaine de l'identification de la langue.

Références

[1] :Said Gadri , Catégorisation Automatique Contextuelle de Documents Semi-structurés Multilingues, Université Ferhat Abbas de Sétif- Sétif 1 Faculté des Sciences Département d'Informatique, 2015 – 2016.

[2]: BENALIA Soumia et OTMANI Marwa, Identification des Langues proches et langueslointaines (Latin versus Arabe), diplôme de Master, Université Saad DAHLAB - Blida 1Faculté des sciences, 2020/2021.

[3]: M. L. M. Z. T. B. TS Jauhiainen, «Automatic Language Identification in Texts: A Survey» Journal of Artificial Intelligence Research, vol. 65, n°1., p. 675–782,2019.

[4] G.-I. Kikui, «Identifying the Coding System and Language of Online Documents onthe Internet,» chez COLING '96, Copenhagen, Denmark, In Proceedings of the 16thInternational Conference on Computational, 1996, p. pp. 652–657.

[5]: T. S. M. T. O. &. W.-H. C. Mandl, Language Identification in Multi-lingual Web-Documents, pp. 153–163, Klagenfurt, Austria: In Proceedings of the 11thInternational Conference on Applications of Natural Language to InformationSystems (NLDB 2006), 2006.

[6]: V. S. P. P. C. &. K. A. Simaki, Identifying the Authors' National Variety of Englishin Social Media Texts, In Proceedings of the International Conference RecentAdvances in Natural Language Processing (RANLP 2017), pp. 671–678: Varna,Bulgaria, 2017.

[7]: P. Henrich, Language Identification for the Automatic Grapheme-to-phonemeConversion of Foreign Words in a German Text-to-speech System., pp. 2220–2223,Paris, France: In First European Conference on Speech Communication andTechnology, 1989.

[8]: S. Langer, Natural Languages and the World Wide Web, .: Bulletin de Linguistique,2001.

[9]: L. G. Windisch, Language identification using global statistics of natural languages,.: Proceedings of the 2nd Romanian-Hungarian Joint Symposium on AppliedComputational Intelligence (SACI), 2005.

[10]: Said K., Abdelouahab M.: An Effective Method to Recognize the Language of a Text in a collection of Multilingual Documents, 10th International Conference on Electronics, Computer and Computation ICECCO 2013, IEEE conference, TurgutOzal University, Ankara, Turkey, 07-08 November 2013..

[11] Said K., Abdelouahab M., Linda B-F. : Language Identification: A New Fast Algorithm to Identify the Language of a Text in a Multilingual Corpus, The 4thInternational Conference on Multimedia Computing and Systems ICMCS'14, IEEE Conference, University of Marrakesh, Marrakesh, Morocco, 14-16 Avril 2014.

[11]:K. K. e. al, Text Classification Algorithms: A Survey, Charlottes ville , USA,:Department of Systems and Information Engineering, University of Virginia,230avril 2019.

[12]: BENALIA Soumia et OTMANI Marwa,Identification des Langues proches et langueslointaines (Latin versus Arabe), diplôme de Master, Université Saad DAHLAB - Blida 1Faculté des sciences, 2020/2021.

[13]:D.D.Lewis, An evaluation of phrasal and clustered representations on a textcategorization task, university of chicago: Center for information and languagestudies, 1992.

[14] S. M. a. S. Scott, Feature engineering for text classification, .: presented atInternational Conference On Machine Learning, , 1999.

[15]:PORTER M.F. An algorithm for su_x stripping. Program, 1980, 14(3), pp. 130{137, doi: 10. 1108/eb046814.

[16]: PRATIKSHA P.Y., GAWANDE S.H. A Comparative Study on Di_erent Types of Approachesto Text Categorization. International Journal of Machine Learning and Computing,2012, 2(4), p. 423.

[17]: RAJMAN M., LEBART L. Similarit_es pour données textuelles. In: 4th international conferenceon statistical analysis of textual data (JADT'98), Nice, France. 1998, pp. 545{555.[In French].

[18]:S. Gadri, Moussaoui. APPLICATION OF A NEW SET OF PSEUDO-DISTANCES IN DOCUMENTS CATEGORIZATION.

[19] H. Shimodaira, Text Classification using Naive Bayes Hiroshi Shimodaira,Édimbourg: University of Edinburgh., January-March 2020.

[20]:W. B. &. T. J. M. Cavnar, N-Gram-Based Text Categorization, Las Vegas, InProceedings of SDAIR-94, Third Annual Symposium on Document Analysis andInformation, 1994.

[21]:G. J. A. C.Buckley, Automatic query expansion using SMART, ornell University,Ithaca: Department of Computer Science, 1994.

[22] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, DragomirAnguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeperwith convolutions. In Computer Vision and Pattern Recognition, pages 1–9, 2015.

[23] Ross Girshick. Fast R-CNN. In IEEE International Conference on Computer Vision,pages 1440–1448, 2015.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification withdeep convolutional neural networks. In International Conference on Neural InformationProcessing Systems, pages 1097–1105, 2012.

[25] G. Said and N. Erich. Building best predictive models using ml and dl approachesto categorize fashion clothes. In Proceedings of the 19th International Conference onArtificial Intelligence and Soft Computing (ICAISC), Zakopane, Poland, October 2020.Springer. H5-index = 20.

[26] S. Gadri and N.E. Adouane. Efficient traffic signs recognition based on CNN model for self-driving cars. In P. Vasant, I. Zelinka, and G.W. Weber, editors, Intelligent Computing Optimization. ICO 2021, volume 371 of Lecture Notes in Networks and Systems. Springer, 2022.

[27] Shaoqing Ren, Ross Girshick, Ross Girshick, and Jian Sun. Faster r-cnn: Towardsreal-time object detection with region proposal networks. IEEE Transactions on PatternAnalysis & Machine Intelligence, 39(6):1137–1149, 2017.

[28] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-basedfully convolutional networks. In Proceedings of the Neural Information ProcessingSystems (NeurIPS), 2016.

[29] G. Said. Efficient arabic handwritten character recognition based on machine learningand deep learning approaches. Journal of Advanced Research in Dynamical & ControlSystems, 12(07-Special Issue):9–17, August 2020.

[30] Hassina T., Said Gadri, Baya L., Nour El-Houda A., and Nadjet B. Handwrittendigit recognition: Developing an efficient ml and dl model to recognize handwrittendigits. In Proceedings of CNIATI'21 Conference (Conférence Nationale sur l'Intelligence Artificielle et les Technologies de l'Information), University Chadli Ben Djedid, Al-Taref, May 2021.

[31] J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim. Energy efficient routingalgorithm with mobile sink support for wireless sensor networks. Sensors, 19(7):1494,2019.

[32] Z. Tang, X. Ding, Y. Zhong, L. Yang, and K. Li. A self-adaptive bell-lapadula modelbased on model training with historical access logs. IEEE Transactions on InformationForensics & Security, 13(8):2047–2061, Aug 2018.

[33] S. Gadri, S. Chabira, S. Ould Mehieddine, and Kh. Herizi. Sentiment analysis: Developing an efficient model based on machine learning and deep learning approaches. In P. Vasant, I. Zelinka, and G.W. Weber, editors, Intelligent Computing Optimization. ICO 2021, volume 371 of Lecture Notes in Networks and Systems. Springer, 2022.

[34] Said G, Sara O.M, Khadidja H, Safia C. An Efficient System to Predict Customers' Satisfaction on Touristic Services Using ML and DL Approaches. 22rd Arabic International Conference on Information Technology (ACIT 2021), 21-23 Dec, IEEE Conference, Quabos University, Sultanate Oman. Available on: https://ieeexplore.ieee.org/document/9677167.

DOI: 10.1109/ACIT53391.2021.9677167

[35] Said G. Developing an efficient Predictive model based on ML and DL approaches to detect diabetes, The International Journal of Computing and Informatics Informatica, ISSN(p): 0350-5596, Volume-45, Issue-3, 2021, https://www.informatica.si/index.php/informatica/article/view/3041

الملخص:

يعد تحديد اللغة مهمةحاسمة تهدف إلى تحديد لغة نص معين بدقة. تستخدمتقنية Bigrams وخوارزميات التعلم الآلي
((MLوالتعلم العميق ((DL على نطاق واسع لهذه المهمة. تقنية bigram هي نموذج لغوي إحصائي يفحص أزواج من الكلمات المتتالية في النص لتحديد اللغة. يستخدم توزيع تردد bigram لتحديد لغة النص. تُستخدم خوارزميات MLمثل Naive Bayes و Support Vector Machine (SVM)على نطاق واسع لتحديد اللغة. تعمل هذه الخوارزميات من خلال تدريب نموذجعلى مجموعة من البيانات المصنفة التي تحتوي على لغات مختلفة. يتنبأ النموذج بعد ذلك بلغة نص معين بناءً على الأنماط التي تم تعلمها.
تُستخدم أيضًا خوارزميات DL مثلا الشبكة العصبية العميقة ((DNNلتحديد اللغة. تتعلم هذه الخوارزميات الأنماط اللغوية المتأصلة من خلال فحص ميزاتالنص على مستوى الحرفومستوى الكلمة. يستخدمون التمثيلات المكتسبة للتنبؤ بدقة بلغة نص معين.
في الختام،لعبت تقنية bigramوخوارزميات ML و DLدورًا حيويًا في تحديد اللغة وحققت تقدمًا كبيرًا في تحديد لغة نص معين بدقة.

Summary:

Language identification is a crucial task aimed at accurately identifying a specific text language. Bigrams technology and Machine Learning (ML) and Deep Learning (DL) algorithms are widely used for this task. Bigram technology is a statistical linguistic model that examines pairs of consecutive words in the text to determine the language. The distribution of bigram frequency is used to determine the language of the text.ML algorithms such as Naive Bayes and Support Vector Machine (SVM) are widely used for language identification. These algorithms work by training a model on a set of classified data containing different languages. The model then predicts a certain text language based on the patterns it has learned. DL algorithms such as deep neural networks (

DNN) are also used to identify language. These algorithms learn the inherent linguistic patterns by examining text features at the character and word level. They use acquired representations to predict a certain text language accurately.

In conclusion, bigram technology and ML and DL algorithms have played a vital role in language identification and have made significant progress in accurately identifying a specific text language.

Résumé:

La détermination de la langue est une tâche cruciale visant à identifier avec précision une langue spécifique dans un texte. La technique des bigrammes et les algorithmes d'apprentissage automatique (ML) et d'apprentissage profond (DL) sont largement utilisés pour cette tâche. La technique des bigrammes est un modèle linguistique statistique qui examine les paires de mots consécutifs dans le texte pour déterminer la langue. Il utilise la distribution de fréquence des bigrammes pour déterminer la langue du texte.Des algorithmes ML tels que Naive Bayes et Support Vector Machine (SVM) sont largement utilisés pour déterminer la langue. Ces algorithmes fonctionnent en entraînant un modèle sur un ensemble de données classées contenant différentes langues. Le modèle prédit ensuite la langue d'un texte spécifique en fonction des modèles qu'il a appris.Des algorithmes DL tels que les réseaux de neurones profonds (dNN) sont également utilisés pour déterminer la langue. Ces algorithmes apprennent les modèles linguistiques inhérents en examinant les caractéristiques du texte au niveau des caractères et des mots. Ils utilisent les représentations acquises pour prédire avec précision la langue d'un texte spécifique.En conclusion, la technique des bigrammes et les algorithmes ML et DL ont joué un rôle crucial dans la détermination de la langue et ont réalisé des progrès significatifs dans l'identification précise d'une langue spécifique dans un texte.