République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma-

Faculté des Mathématiques, d'Informatique et des Sciences de la matière
Département d'Informatique

Mémoire de projet de fin d'étude Master

Filière : Informatique

Option : Système Informatique

Thème :

Une méthode hybride basée sur l'information mutuelle et les algorithmes génétiques pour la sélection des attributs

Encadré Par :
Dr. Farek Lazhar

Présenté Par :
Mébaki Houneida

Juin 2023

Remerciements

Je voudrais, dans un premier temps, remercier le bon Dieu tout-puissant de m'avoir donné le courage et la volonté de réaliser ce projet.

Mes remerciements les plus chaleureux vont à mon directeur de mémoire, Dr. Farek Lazhar, pour sa patience, sa disponibilité et surtout ses judicieux conseils qui m'ont permis d'améliorer la qualité de ce travail.

Enfin, mes remerciements s'adressent à tous les professeurs du département d'informatique de l'université du 8 mai 1945 de Guelma.

Dédicaces

Je dédie cet ouvrage

À mes parents pour leur sacrifice et leur soutien indéfectible tout au long de ces années d'études.

Que ce travail soit l'accomplissement de vos vœux les plus chers, et le fruit de votre soutien infaillible.

Je dédie également ce travail à mon frère, ma sœur et à ceux qui ont partagé avec moi tous les moments d'émotion lors de sa réalisation. Ils m'ont chaleureusement soutenu et encouragé tout au long de mon parcours.

Je tiens à remercier mes chers amis d'être toujours là pour moi.

Je dédie ce travail à tous ceux que j'aime.

Résumé

La sélection de caractéristiques est une étape cruciale dans le processus d'apprentissage automatique, visant à partir d'un ensemble de données d'origine, à identifier et sélectionner les caractéristiques les plus informatives. Parmi les techniques utilisées dans ce processus, on retrouve MI, CH2, IGI, etc. Malgré leur efficacité, ces techniques souffrent de l'inconvénient de la redondance, ce qui entraîne une faible performance du modèle de classification.

Dans ce travail, nous avons opté pour une approche hybride basée sur la méthode IG et l'algorithme génétique. Tout d'abord, nous utilisons l'IG pour évaluer la relation entre chaque caractéristique et la variable de classe. Les caractéristiques ayant un score IG fort sont considérées comme plus discriminantes. Ensuite, nous utilisons un algorithme génétique pour effectuer une recherche dans l'espace des caractéristiques sélectionnées par l'IG et trouver un sous-ensemble optimal en utilisant un ensemble d'opérations telles que la sélection, le croisement et la mutation.

Les résultats des expérimentations confirment que notre méthode hybride a atteint notre objectif en améliorant la performance, en réduisant considérablement la redondance, et en surpassant les autres méthodes.

Mots clés : Sélection de caractéristique, redondance, classification, texte, terme, entropie, algorithme génétique, fonction de fitness, croisement, mutation.

Absract

Feature selection is a crucial step in the machine learning process, aiming to identify and select the most informative features from an original dataset. Among the techniques used in this process, we find MI, CH2, IGI, etc. Despite their effectiveness, these techniques suffer from the drawback of redundancy, resulting in poor classification model performance.

In this work, we opted for a hybrid approach based on the IG method and genetic algorithm. Firstly, we use IG to evaluate the relationship between each feature and the class variable. Features with high IG scores are considered more discriminative. Then, we employ a genetic algorithm to search within the space of features selected by IG and find an optimal subset using operations such as selection, crossover, and mutation.

The experimental results confirm that our hybrid method has achieved our objective by improving performance, significantly reducing redundancy, and outperforming other methods.

Keywords: Feature selection, redundancy, classification, text, term, entropy, genetic algorithm, fitness function, crossover, mutation.

Table des Matières

Résumé **III**

Abstract **IV**

Liste des figures **X**

Liste des tableaux **XII**

Abréviations et acronyms **XIII**

Introduction Générale **1**

1. Problématique....1
2. Organisation du Mémoire....2

Chapitre 01 : Classification des textes **3**

1.1 Introduction....4

1.2 Définition....4

1.3 Processus de classification....4

1.4 Classification supervisée.... ..5

1.5 Types de classification supervisée....5

1.5.1 Classification binaire....5

1.5.2 Classification Multi-label....5

1.5.3 Classification Multi-classes.... 6

1.6 Évaluation des modèles de classification....6

1.6.1 Matrice de confusion....6

1.6.2 Accuracy.... 7

1.6.3 Précision.... 7

1.6.4 Rappel.... 8

1.6.5 F1-Score.... 8

1.7 Méthodes de pondération.... 8

1.7.1 Sac de mots (BOW)....8

1.7.2 TF-IDF..... 9
1.8 Algorithmes de classification..... 10
1.8.1 Support Vector Machine (SVM)..... 10
1.8.2 k-Nearest Neighbor (k-NN).....12
1.8.3 Arbre de décision (AD).....13
1.8.4 Naive Bayes (NB)..... 14
1.8.5 Les Réseaux de neurones artificiels..... ..15
1.9 Applications de classification de textes..... 16
1.10 Les difficultés de classification de textes..... ..16
1.11 Conclusion..... 17
Chapitre 02 : Sélection des features **19**
2.1 Introduction..... ..20
2.2 Définition..... 20
2.3 Processus de sélection d'attributs..... 20
2.4 Objectifs de la sélection des features..... 21
2.5 Métriques de sélection pour la classification de textes..... 21
2.5.1 Gain d'information (IG).....22
2.5.2 Indice de Gini (GI)..... ..22
2.5.3 Chi-Square.....22
2.5.4 Fréquence des documents..... 23
2.5.5 Information Mutuelle (MI).....23
2.6 Approches de sélection des features.....23
2.6.1 Approche filtre..... 24
2.6.2 Approche enveloppe.....24
2.6.3 Approche intégrée.....25
2.6.4 Approche hybride.....26

2.7 Conclusion..... 27

Chapitre 03 : Les algorithmes génétiques **28**

3.1 Introduction..... ..29

3.2 L'évolution des algorithmes génétiques..... .29

3.3 Définition..... 29

3.4 Espace de recherche..... 30

3.5 Terminologie..... 30

3.6 Les composants..... .31

1. Le codage.....31

2. Initialisation.....31

3. La fonction de fitness.....31

4. Les opérateurs..... 31

5. Les paramètres..... 32

6. Critère d'arrêt..... 32

3.7 Le fonctionnement d'algorithme génétique..... 32

3.8 Variantes des GAs..... .33

3.8.1 Le codage..... .33

3.8.1.1 Codage binaire..... 33

3.8.1.2 Codage réel..... 33

3.8.2 La fonction de fitness..... .34

3.8.3 Initialisation..... 34

3.8.4 La sélection..... 34

3.8.4.1 La sélection par classement..... .35

3.8.4.2 La sélection par la roulette..... 35

3.8.4.3 La sélection par tournoi..... 36

3.8.4.4 L'élitisme..... 36

3.8.5 Le croisement..... 36

3.8.5.1 Croisement en 1-point..... 37

3.8.5.2 Croisement en 2-points..... .37

3.8.5.3 Croisement en n-points..... ...37

3.8.5.4 Croisement uniforme..... ...37

3.8.5.5 Croisement réel..... ...38

a) Ordre de base cyclique..... 38

b) Croisement d'ordre maximal..... 39

3.8.6 Mutation..... 39

3.8.6.1 Mutation en codage binaire..... .39

3.8.6.2 Mutation en codage réel.....39

a) L'opérateur d'inversion simple.....40

b) L'opérateur d'insertion.....40

c) L'opérateur d'échange réciproque.....41

3.9 Valeurs des paramètres.....41

3.10 Les avantages et les limites.....42

3.11 Domaines d'application.....43

3.12 Conclusion.....43

Chapitre 04 : La méthode proposée 44

4.1 Introduction.....45

4.2 Impact des features redondantes sur les performances et le temps d'exécution du classifieur.....45

4.3 La méthode proposée.....46

4.3.1 Prétraitement.....47

a) Elimination des mots vides.....47

b) La conversion en minuscules.....48

c) Suppression des caractères spéciaux.....48

d) La tokénisation.....48

e) La lemmatisation.....48

4.3.2 La construction du vocabulaire des mots uniques.....48

4.3.3 Sélection basée sur le gain d'information.....48

4.3.4 Sélection basée sur les algorithmes génétiques.....51

4.5 Conclusion.....57

Chapitre 05 : Expérimentation et évaluation 59

5.1 Introduction.....60

5.2 Description d'environnements et de bibliothèques.....60

5.2.1 L'environnement utilisé.....60

5.2.2 Les bibliothèques nécessaires..... .60

5.3 Présentation et prétraitement des jeux de données.....61

5.3.1 Description des jeux de données.....61

5.3.2 Prétraitement.....63

5.4 Classification..... .64

5.4.1 Classification sans sélection de features.....64

5.4.2 Classification avec sélection des features par MI, CH2,IGI et IG.....65

5.4.3 Sélection avec l'algorithme génétique.....66

5.5 Discussion.....68

5.6 Conclusion.....72

Conclusion générale et perspectives 73

Bibliographie & Webographie 74

Liste des Figures

1.1 Processus de la classification des textes 5
1.2 Un exemple de cas linéairement séparable. 11
1.3 Exemple d'arbre de décision.....14

2.1 Présentation du processus de sélection des attributs.....21
2.2 Principe de l'approche filter…….....24
2.3 Principe de l'approche enveloppe.....25
2.4 Principe de l'approche intégrée..... 26

3.1 Exemple d'un paysage de fitness 30
3.2 Le diagramme génétique.....32
3.3 La sélection par la roulette.....36
3.4 Croisement en 1-point.....37
3.5 Croisement uniforme.....38
3.6 Croisement d'ordre de base cyclique.....38
3.7 Croisement d'ordre maximal.....39
3.8 Mutation par inversion simple..... 40
3.9 Mutation par insertion..... 40
3.10 Mutation par échange réciproque..... 41

4.1 Exemple illustratif sur l'initialisation.....52
4.2 Sélection des individus pour la reproduction..... 54
4.3 Un exemple illustratif sur le croisement..... ..55
4.4 Un exemple illustratif sur la mutation.....56

5.1 Un aperçu du dataset Restaurant..... .62
5.2 Un aperçu du dataset Fake News.....62
5.3 Un aperçu du dataset Restaurant après prétraitement.....64
5.4 Un aperçu du dataset Fake News après prétraitement.....64
5.5 Résultats de classification de Fake avec NB..... ..68
5.6 Résultats de classification de Fake avec SVM.....68

5.7 Résultats de classification de Restaurant avec NB.....69
5.8 Résultats de classification de Restaurant avec SVM.....69
5.9 Résultats de classification de Fake News avec NB (MI-GA).....69
5.10 Résultats de classification de Fake News avec SVM (MI-GA)..... ...69
5.11 Résultats de classification de Fake News avec NB (CH2-GA)..... 69
5.12 Résultats de classification de Fake News avec SVM (CH2-GA).....69
5.13 Résultats de classification de Fake News avec NB (IGI-GA).....70
5.14 Résultats de classification de Fake News avec SVM (IGI-GA) 70
5.15 Résultats de classification de Fake News avec NB (IG-GA).....70
5.16 Résultats de classification de Fake News avec SVM (IG-GA)..... 70
5.17 Résultats de classification de Restaurant avec NB (MI-GA)..... 70
5.18 Résultats de classification de Restaurant avec SVM (MI-GA).....70
5.19 Résultats de classification de Restaurant avec NB (CH2-GA).....70
5.20 Résultats de classification de Restaurant avec SVM (CH2-GA)..... 70
5.21 Résultats de classification de Restaurant avec NB (IGI-GA).....71
5.22 Résultats de classification de Restaurant avec SVM (IGI-GA)..... 71
5.23 Résultats de classification de Restaurant avec NB (IG-GA)..... 71
5.24 Résultats de classification de Restaurant avec SVM (IG-GA)..... 71

Liste des Tableaux

1.1 Matrice de confusion.....6

4.1. Paramètres de notre algorithme génétique 52

5.1 Résultats de classification sans sélection de features.....65

5.2 Résultats classification avec sélection de features pour le dataset Fake News 65

5.3 Résultats classification avec sélection de features pour le dataset Restaurant 65

5.4 Évolution de l'Accuracy par génération pour MI, CH2, IGI et IG avec GA pour Fake News..... 66

5.5 Évolution de l'Accuracy par génération pour MI, CH2, IGI et IG avec GA pour Restaurant.....67

5.6 Résultats classification pour (MI, CH2,IGI et IG) avec GA pour Fake News.....67

5.7 Résultats classification pour (MI, CH2,IGI et IG) avec GA pour Restaurant.....68

Abréviations et acronymes

<AD> <Arbre de Décision>

<BOW> <Bag of Words>

<CH2> <Chi-Square>

<DF> <document Frequency>

<GA> <Genetic Algorithm>

<GI> <Gini Index>

<IG> <Information Gain>

<KNN> <K Nearest Neighbor>

<MI> <Mutual Information>

<NB> <Naïve Bayes>

<SVM> <Support Vector Machine>

<TF> <Term Frequency>

<TF-IDF> <Term Frequency-Inverse Document Frequency>

Introduction Générale

1. Problématique

La classification de texte joue un rôle important dans le domaine de l'intelligence artificielle, et elle a considérablement évolué au fil des années pour contribuer à de nombreux domaines variés. Ce processus consiste à attribuer automatiquement des catégories spécifiques ou arbitraires à des données textuelles.

Dans ce domaine, la dimension de l'ensemble de données textuelles collectées est généralement large, ce qui conduit à des performances lentes du classificateur et réduit sa capacité à prédire correctement. Afin d'éviter ce problème, une technique supplémentaire est nécessaire.

Il s'agit de la sélection des caractéristiques (anglais : features). Peu importe le texte, il est constitué d'un ensemble de termes ou de caractéristiques. Ainsi, la sélection des caractéristiques permet d'extraire un sous-ensemble des termes les plus informatifs et d'éliminer ceux qui ne sont pas significatifs pour le problème en question. Ce processus est une étape cruciale pour faciliter la tâche de classification des textes.

Les algorithmes les plus couramment utilisés dans la littérature sont : Information Gain (IG), Chi-Square Test (CH2), Mutual Information (MI) et Imroved Gini Index (IGI). Ce sont des méthodes statistiques basées sur des calculs spécifiques. Ils ont prouvé leur efficacité depuis leurs débuts. Cependant, ils posent un problème majeur : lorsqu'ils sélectionnent des termes, ils peuvent être redondants, ce qui affecte considérablement leurs performances.

Ainsi, notre problème est le suivant : Comment concevoir une méthode qui permet de sélectionner de manière optimale un sous-ensemble de caractéristiques afin d'éviter la redondance, de réduire la dimensionnalité et d'améliorer les performances de notre modèle de classification ?

Pour remédier à ce problème, nous proposons une approche hybride basée sur l'IG et l'algorithme génétique. L'IG est une méthode basée sur le calcul des entropies de classes et des entropies conditionnelles. Quant à l'algorithme génétique, il s'agit d'une méthode de recherche basée sur l'évolution biologique, qui peut être utilisée pour explorer efficacement l'espace de recherche des combinaisons de caractéristiques.

Nous appliquons d'abord l'IG pour extraire les termes les plus pertinents, puis nous utilisons le GA pour optimiser les termes sélectionnés par l'IG, nous obtenons des résultats améliorés en termes de performance de classification et de complexité temporelle.

2. Organisation du Mémoire

Ce travail est composé de 5 chapitres comme suit :

Chapitre 1. **Classification des textes :** Dans ce chapitre, nous présenterons un aperçu général de la classification de textes, en mettant en évidence les différents domaines et problèmes liés à cette étude. Nous détaillerons les différents algorithmes de classification ainsi que les méthodes de pondération les plus répandues.

Chapitre 2. **Sélection des features** : Ce chapitre abordera le processus de sélection des features, en présentant les métriques de sélection les plus utilisées pour la classification de textes, telles que l'IG, l'IGI, CH2 et MI. Nous expliquerons également le fonctionnement des différentes approches de sélection des features, telles que wrapper, filter, hybride et embedded.

Chapitre 3. **Les algorithmes génétiques :** Ce chapitre fournira un aperçu général des algorithmes génétiques, en mettant l'accent sur leur fonctionnement, leurs composants, leurs variantes, ainsi que leurs avantages, limites et domaines d'application.

Chapitre 4. **Conception :** Ce chapitre présentera les étapes conçues pour appliquer notre méthode proposée : **Une méthode hybride basée sur l'information mutuelle et les algorithmes génétiques.**

Chapitre 5. **Expérimentation et évaluation :** Nous évaluons la performance de notre approche hybride en utilisant deux classificateurs Naïve Bayes (NB) et Support Vector Machine (SVM). Nous comparerons les résultats obtenus avec ceux des autres méthodes.

Chapitre 1

Classification des textes.

1.1 Introduction

La classification de texte est une tâche fondamentale de la science des données et du traitement automatique du langage naturel. Elle consiste à attribuer une catégorie ou un label à un document texte en fonction de son contenu. Les applications de la classification de texte sont nombreuses, allant de la catégorisation de courriels spam aux analyses de sentiments des commentaires en ligne.

Dans ce chapitre, nous explorerons le processus de classification de texte, ses différents types, ses algorithmes, ses métriques d'évaluation ainsi que ses méthodes de représentation.

1.2 Définition

La classification de texte consiste à attribuer automatiquement des documents textuels (tels que des documents en texte brut et des pages Web) à des catégories prédéfinies en fonction de leur contenu. En termes formels, la classification de texte fonctionne sur un espace d'instances X où chaque instance est un document d, et un ensemble fixe de classes C = {C1,C2,...,C|C|} où |C| est le nombre de classes. Étant donné un ensemble D de documents d'entraînement ⟨d, Ci ⟩ ou ⟨d, Ci ⟩ ∈ X × C, en utilisant une méthode d'apprentissage ou un algorithme d'apprentissage, l'objectif de la classification de documents est d'apprendre un classificateur ou une fonction de classification γ qui mappe les instances aux classes: γ: X → C [1].

Les trois catégories d'apprentissage automatique sont : supervisé, non supervisé et semi-supervisé.

1.3 Processus de classification

Lorsqu'on construit un système de classification de texte, plusieurs étapes sont nécessaires. Tout d'abord, il faut collecter ou créer un ensemble de données étiquetées adapté à la tâche. Ensuite, on divise l'ensemble de données en ensembles d'entraînement, de validation et de test, en choisissant les métriques d'évaluation adaptées. Puis, on transforme le texte brut en vecteurs de caractéristiques et on entraîne un classificateur à l'aide des vecteurs de caractéristiques et des étiquettes correspondantes de l'ensemble d'entraînement. On évalue les performances du modèle sur l'ensemble de test à l'aide des métriques d'évaluation choisies. Enfin, on déploie le modèle pour répondre aux besoins du monde réel et on surveille ses performances [2].

Le schéma ci-dessous montre les étapes pour construire un système de classification de texte :

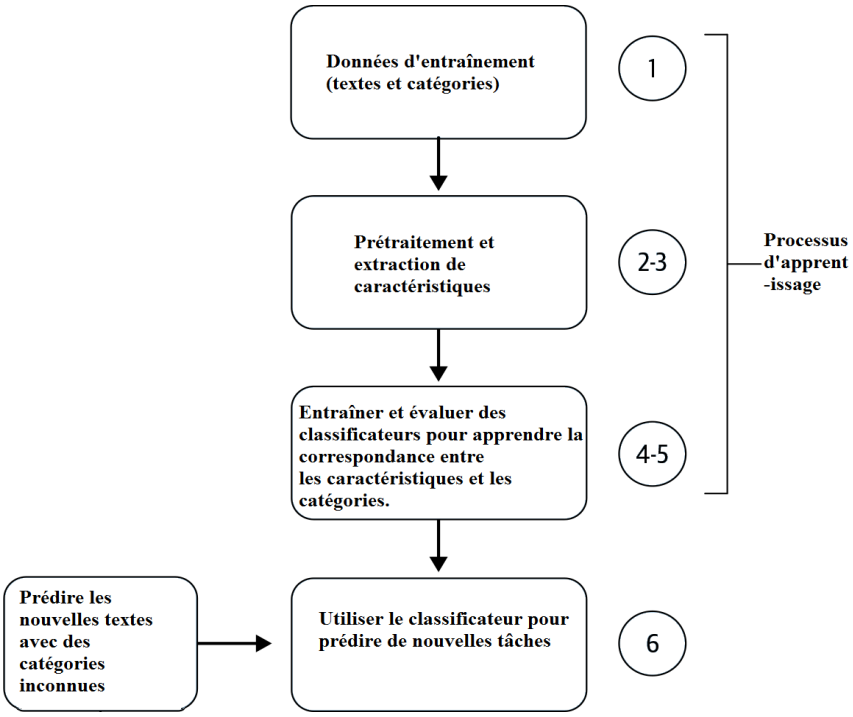

Figure 1.1-Processus de la classification des textes [2].

1.4 Classification supervisée

Cette technique implique de fournir des exemples d'entrée avec des étiquettes (c'est-à-dire des réponses connues) au modèle, afin qu'il puisse apprendre à prédire de manière précise les étiquettes pour de nouvelles données similaires [3]. Cette méthode de classification est utilisée dans différents domaines tels que la détection de fraudes ou encore la prédiction de la demande de produits.

1.5 Types de classification supervisée

Il existe plusieurs types d'algorithmes de classification. Dans cette section, nous allons explorer les types les plus courants.

1.5.1 Classification binaire

Dans la classification binaire, il y a deux classes possibles. Cette classification consiste à attribuer une nouvelle observation dans l'une de ces classes. Prenons l'exemple d'un diagnostic médical pour une maladie inconnue. À partir d'un ensemble de tests médicaux pour évaluer la présence ou l'absence de la maladie chez un patient.

1.5.2 Classification Multi-label

Les méthodes de classification multi-label peuvent être catégorisées en deux groupes :

- **Les méthodes de transformation de problème :** Ce groupe de méthodes est indépendant de l'algorithme. Elles transforment la tâche de classification multi-label en une ou plusieurs tâches de classification à mono-label, de régression ou de classement de label.

- **Les méthodes adaptées :** utilisent des algorithmes d'apprentissage spécifiques afin de gérer directement les données multi-label [4].

1.5.3 Classification Multi-classes

La classification de texte multi-classe est une tâche de classification de texte avec plus de deux classes. Chaque Données textuelles peut être classée dans l'une des classes. Cependant, un exemple de données ne peut pas appartenir à plus d'une classe simultanément.

Par exemple, un modèle qui classifie les titres de nouvelles dans des catégories de nouvelles. Les catégories peuvent être les sports, la technologie et la politique…etc [5].

1.6 Évaluation des modèles de classification

Les métriques de classification sont couramment utilisées pour comparer les performances de différents classificateurs et ensembles de caractéristiques, et ces métriques sont généralement basées sur la matrice de confusion binaire. La sélection de métriques de performance de classification appropriées est cruciale pour évaluer l'efficacité des procédures de classification [6].

1.6.1 Matrice de confusion

La matrice de confusion est un outil qui permet d'évaluer les performances d'un modèle de classification, en d'autres termes, elle est utilisée pour évaluer la qualité des prédictions du modèle en examinant les résultats de chaque classe.

		valeurs réelles	
		V	N
Valeurs prédites	V	VP	FP
	N	FN	VN

Table 1.1- Matrice de confusion [7].

Où :

VP : Vrai positif (True Positive) : C'est le cas où le modèle prédit correctement une classe positive.

FP : Faux positif (False Positive) : C'est le cas où le modèle prédit à tort une classe positive.

VN : Vrai négatif (True Negative) : C'est le cas où le modèle prédit correctement une classe négative.

FN : Faux négatif (False Negative) : C'est le cas où le modèle prédit à tort une classe négative.

1.6.2 Accuracy

Accuracy également appelée (taux du succès) peut être défini comme la fraction de prédictions correctes faites par un classificateur sur l'ensemble de test. Accuracy est l'opposé du taux d'erreur et les deux taux donnent les mêmes informations sur la force ou la faiblesse d'un classificateur.

Un inconvénient d'Accuracy et du taux d'erreur est qu'ils ne donnent pas d'informations sur la performance d'un classificateur pour chaque classe séparément, mais fournissent plutôt une mesure globale de la performance pour l'ensemble de test [7].

$$accuracy = \frac{VP+VN}{VP+VN+FP+FN} \qquad (1.1)$$

$$taux\ d'erreur = \frac{FP+FN}{VP+VN+FP+FN} \qquad (1.2)$$

1.6.3 Précision

La précision est le rapport entre les vrais positifs et tous les positifs observés. Tous les positifs ici comprennent les vrais positifs observés, ainsi que certains vrais négatifs observés à tort comme des positifs [7].

$$précision = \frac{VP}{VP+FP} \qquad (1.3)$$

1.6.4 Rappel

Le Rappel, appelé également la Sensibilité (Recall en anglais) est la capacité d'un modèle à identifier correctement les vrais positifs.

$$rappel = \frac{VP}{VP+FN} \qquad (1.4)$$

Un haut rappel signifie que les vrais positifs observés sont relativement plus élevés que les faux négatifs observés, et un faible rappel signifie l'opposé [7].

1.6.5 F1-Score

Est une mesure qui combine à la fois la précision et le rappel, une mesure F1 élevée est un indicateur que le modèle de classification a une bonne précision et un bon rappel.

$$F1 - Score = 2 \times \frac{precision \times rappel}{precision + rappel} \qquad (1.5)$$

1.7 Méthodes de pondération

La pondération des termes est une étape cruciale dans le processus de prétraitement de données en classification de texte. Qui consiste à attribuer des poids adaptés à chaque terme présent dans les documents afin d'améliorer les performances des classificateurs.

La représentation de texte est une étape cruciale dans la classification de texte et consiste à convertir le contenu d'un document en un format particulier. Une approche courante de la représentation de texte est le modèle d'espace vectoriel BOW, qui représente chaque document sous la forme d'un vecteur 2D dans lequel la première dimension représente les termes à l'intérieur des documents, tandis que l'autre dimension est utilisée comme référence pour les documents [8].

1.7.1 Sac de mots (BOW)

Le modèle sac de mots (bag of words en anglais) est une méthode simplifiée couramment utilisée en traitement du langage naturel et en recherche d'informations. Il consiste à représenter un texte comme une collection désordonnée de ses mots, en ignorant la grammaire et l'ordre des mots. Pour la classification de texte, chaque mot dans un document se voit attribuer un poids en fonction de sa fréquence dans le document et de sa fréquence entre différents documents. En conséquence, les mots et leurs poids forment le modèle sac de mots [9].

Soit l'exemple suivant, il s'agit de deux documents D1 et D2 :

D1 : i love dogs, dogs are my favorite animals.

D2 : i hate spiders,spiders are creepy.

Le vocabulaire dans ce cas serait : [i : 1, love : 2, dogs : 3, are : 4 , my: 5, favorite : 6, animals : 7, hate : 8, spiders : 9, creepy : 10].

Le modèle de BOW pour ces deux phrases avec 10 mots dans le dictionnaire et une fréquence de 2 pour chaque mot serait le suivant :

i love dogs, dogs are my favorite animals : [1,1,2,1,1,1,1,0,0,0]

i hate spiders, spiders are creepy.: [1, 0, 0, 1, 0, 0, 0, 1, 2, 1]

1.7.2 TF-IDF

Le TF-IDF est largement utilisé dans les domaines de la recherche d'information. Il est utilisé pour extraire les mots clés des documents, calculer les degrés de similarité entre les documents, décider du classement de recherche. Le TF dans TF-IDF signifie l'occurrence de mots spécifiques dans les documents. Les mots ayant une valeur TF élevée sont importants dans les documents. En revanche, le DF indique combien de fois un mot spécifique apparaît dans la collection de documents. Il calcule l'occurrence du mot dans plusieurs documents, pas dans un seul document. Les mots ayant une valeur DF élevée n'ont pas d'importance car ils apparaissent couramment dans tous les documents. En conséquence, l'IDF qui est l'inverse du DF est utilisé pour mesurer l'importance des mots dans tous les documents. Des valeurs IDF élevées signifient des mots rares dans tous les documents, ce qui augmente leur importance.

Les étapes pour calculer le TF-IDF sont les suivantes :

TF:

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \qquad (1.6)$$

$\boldsymbol{n_{i,j}}$ représente le nombre d'occurrences du terme $\boldsymbol{t_i}$ dans le document $\boldsymbol{d_j}$, $\boldsymbol{n_{k,j}}$ représente le nombre total d'occurrences de tous les mots dans le document $\boldsymbol{d_j}$. **K** et **D** représentent respectivement le nombre de termes clés et de documents.

DF:

Le DF est calculé en divisant le nombre total de documents par le nombre de documents qui contiennent un mot clé spécifique.

$$DF_{i,j} = \frac{|d_j \in D : t_j \in d_j|}{|D|} \qquad (1.7)$$

Ou $|D|$ représente le nombre total de documents, $|d_j \in D : t_j \in d_j|$ représente le nombre de documents dans lesquels le terme clé t_j apparaît.

TF-IDF:

$$IDF_{i,j} = log \frac{|D|}{|d_j \in D : t_j \in d_j|} \qquad (1.8)$$

En utilisant les équations (1.6) et (1.8), le TF-IDF est défini comme suit :

$$TF - IDF = TF \times IDF \qquad (1.9)$$

La valeur TF-IDF augmente lorsqu'un terme spécifique a une fréquence élevée dans un document et que la fréquence des documents qui contiennent le terme parmi l'ensemble des documents est faible. Ce principe peut être utilisé pour trouver les termes qui apparaissent fréquemment dans les documents. Le calcul de TF-IDF nous permet de trouver quels termes sont importants dans chaque document [10].

L'un des limites de la technique TF-IDF est que le calcul de l'espace vectoriel peut être lent si le document est de grande longueur [11].

1.8 Algorithmes de classification

Parmi les algorithmes d'apprentissage les plus couramment utilisés pour étudier le problème de classification de texte nous citons :

1.8.1 Support Vector Machine (SVM)

Les classificateurs SVM divisent l'espace de données en utilisant des limites linéaires ou non linéaires entre les différentes classes. L'objectif principal de ces classificateurs est de trouver les frontières optimales entre les classes et ainsi classifier les données en fonction de leur groupe d'appartenance [12].

SVM, dans sa forme de base, est un classificateur binaire linéaire qui identifie une frontière unique entre deux classes. Le SVM linéaire suppose que les données multidimensionnelles sont linéairement séparables dans l'espace d'entrée. En particulier, les SVM déterminent un hyperplan optimal (une ligne dans le cas le plus simple) pour séparer l'ensemble de données en un nombre discret de classes prédéfinies en utilisant les données d'entraînement. Pour maximiser la séparation ou la marge, les SVM utilisent une partie de l'échantillon d'entraînement qui se trouve le plus proche de la frontière de décision optimale dans l'espace des caractéristiques, agissant comme des vecteurs de support [13].

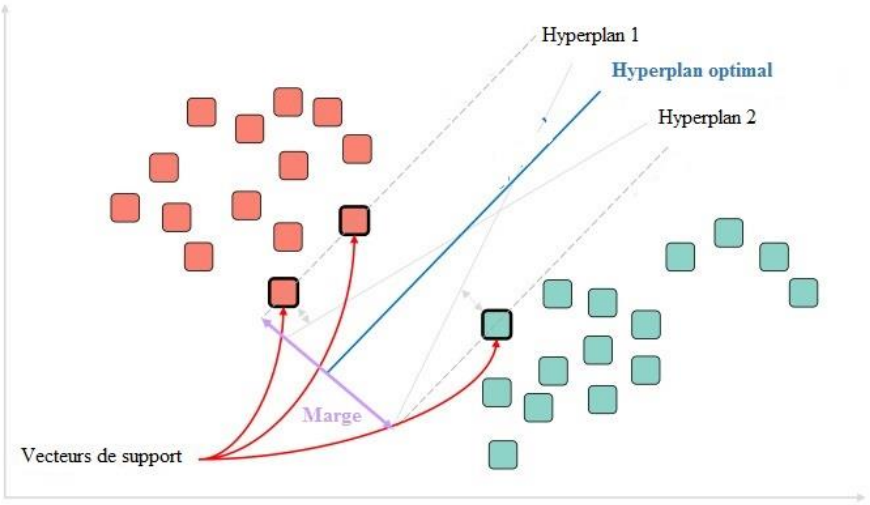

Figure 1.2-Un exemple de cas linéairement séparable [13].

Lorsque des objets linéairement non séparables sont présents, une transformation non linéaire est utilisée pour projeter les données dans un espace de dimensions supérieures. Cette méthode permet de trouver un hyperplan de séparation linéaire dans cet espace en utilisant une fonction de noyau. La fonction de noyau évite de calculer directement les similarités entre les vecteurs dans l'espace de dimensions supérieures en dérivant plutôt ces similarités dans l'espace de dimensions inférieures d'origine [14].

Les avantages :

- Ils sont robustes dans les espaces de grande dimension. Le sur-apprentissage n'affecte pas tellement le calcul de la marge de décision finale.

- Toute caractéristique est importante. Même certaines caractéristiques considérées comme sans importance ont été trouvées utiles lors du calcul de la marge.

- La plupart des problèmes de catégorisation de texte sont linéairement séparables [15].

Les limites :

- Temps d'entraînement important.

- Plus de caractéristiques, plus de complexités.

- Mauvaises performances en cas de bruit élevé.

1.8.2 k-Nearest Neighbor (k-NN)

K-NN est un algorithme de classification supervisée qui utilise les échantillons d'entraînement eux-mêmes pour générer des règles de classification. Il prédit la catégorie d'un échantillon de test en se basant sur les K échantillons d'entraînement qui sont les plus proches voisins de l'échantillon de test, et le classe dans la catégorie ayant la probabilité la plus élevée.

Pour classer un nouveau document dans l'une des j catégories d'entraînement, tout d'abord, nous convertissons le nouveau document en un vecteur de caractéristiques de m dimensions, où m est le nombre de caractéristiques utilisées pour l'entraînement. Ensuite, nous calculons la similarité entre le nouveau document et chacun des échantillons d'entraînement. Nous choisissons k échantillons d'entraînement ayant la similarité la plus élevée avec le nouveau document, qui constitueront la collection K-NN du nouveau document. Ensuite, nous calculons la probabilité que le nouveau document appartienne à chaque catégorie en utilisant les échantillons K-NN. Enfin, nous classons le nouveau document dans la catégorie ayant la probabilité la plus élevée [16].

Les trois paramètres principaux de l'algorithme :

Métrique de similitude / distance : Métrique de distance calcule la différence d'instance pour la similarité. Il est crucial de choisir une métrique de distance appropriée pour obtenir une performance efficace dans le modèle de classification de texte.

K-value : détermine la taille du voisinage et aide à la détermination de la classe.

La probabilité de classe : est basée sur un vote pour assigner une instance de données à une classe [17].

Les avantages :

- L'entraînement est très rapide**.**

- Simple et facile à apprendre.

- Efficace si les données d'entraînement sont nombreuses [18].

Les limites :

- Biaisé par la valeur de k.

- Complexité de calcul**.**

- Limitation de mémoire

- Facilement trompé par les attributs non pertinents [18].

1.8.3 Arbre de décision (AD)

Les arbres de décision (AD) sont un type de modèle prédictif utilisé dans l'exploration de données supervisées. Ils représentent les données sous forme d'une structure arborescente, où chaque nœud interne représente un test sur une caractéristique, chaque branche indique le résultat du test et chaque nœud feuille représente l'étiquette de classe. Le nœud racine n'a pas d'arête entrante et initialement, toutes les instances de données sont au nœud racine. L'arbre atteint la classification en divisant de manière récursive les branches de l'arbre en fonction d'un test d'une caractéristique de données, jusqu'à ce que le dernier niveau soit atteint, où toutes les instances de données dans un nœud appartiennent à une seule classe.

La taille d'un arbre de décision peut être réduite en utilisant une technique appelée "pruning". Cette technique consiste à supprimer les sous-arbres qui reflètent du bruit ou des valeurs aberrantes. Une fois l'arbre complet généré, un algorithme est appliqué pour vérifier s'il existe des sous-arbres répétés. Si tel est le cas, l'arbre de décision est pruned. Cela permet d'obtenir des arbres de décision plus rapides et plus fiables [19].

Il existe différents inducteurs d'arbres de décision descendant tels que ID3, C4.5 et CART . Certains se composent de deux phases conceptuelles: prunning et la croissance (C4.5 et CART). D'autres inducteurs ne réalisent que la phase de croissance [20].

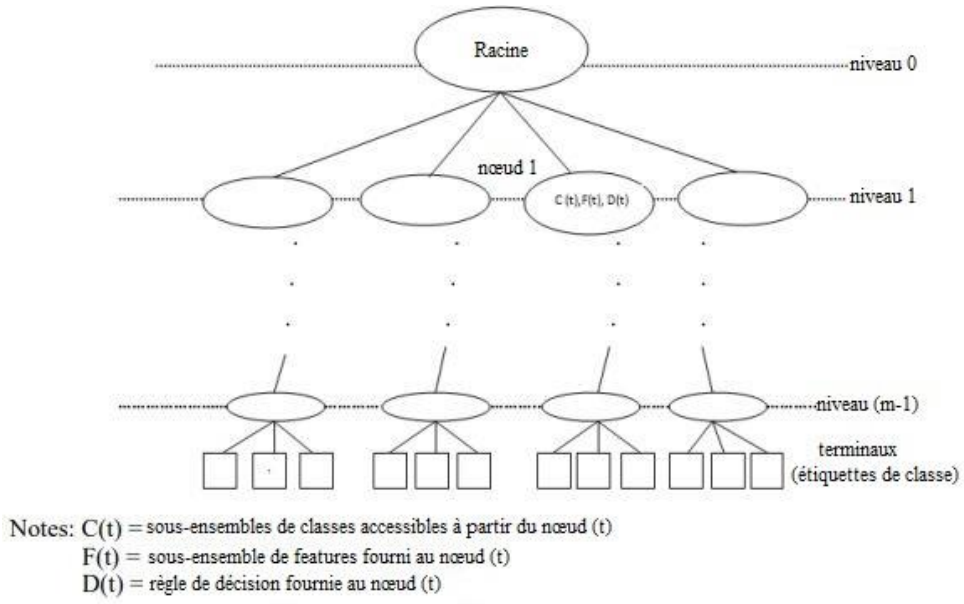

Figure 1.3- Exemple d'arbre de décision [19].

Avantages : Ils ont les avantages suivants :

- Les (ADs) peuvent traiter des types de données numériques, catégoriels et textuels.
- Ils peuvent traiter des ensembles de données erronés et des valeurs manquantes.
- Les (ADs) ont une haute performance avec peu d'effort de calcul. Ils sont utiles pour le regroupement, la sélection de caractéristiques, la régression et la classification.

Les limites : Certaines des limites sont :

- **La taille :** la taille des arbres de décision qui peut être complexe et consommatrice de temps. **L'instabilité :** tout petit changement peut causer des modifications majeures dans l'arbre de décision, nécessitant un redessinage complet.
- Les (ADs) ont des difficultés avec les grands ensembles de données contenant des millions d'attributs dans les applications de fouille de données [19].

1.8.4 Naive Bayes (NB)

L'algorithme Naive Bayes est une méthode d'apprentissage automatique basée sur les probabilités conditionnelles. Il utilise le théorème de Bayes pour calculer la probabilité qu'une instance appartienne à une classe. Les classificateurs bayésiens trouvent la distribution des valeurs d'attribut pour chaque classe dans les données d'entraînement, puis utilisent ces informations pour estimer la probabilité qu'une instance appartienne à chaque classe. Le

classificateur Naive Bayes suppose que les caractéristiques sont indépendantes les unes des autres au sein de chaque classe, une forte indépendance entre les caractéristiques signifie qu'une caractéristique dans une donnée a le même poids important, sans être liée à une autre [21], mais il fonctionne toujours bien même lorsque cette hypothèse n'est pas valable.

$$p(c_j|d) = \frac{p(d|c_j)p(c_j)}{p(d)} \qquad (1.10)$$

Où $p(c_j|d)$est la probabilité de générer l'instance d étant donnée la classe c_j, $p(c_j)$est la probabilité d'occurrence de la classe c_j , et $p(d)$est la probabilité que l'instance d se produise. Il apprend à partir des données d'entraînement et prédit la classe avec la plus forte probabilité a posteriori en supposant que tous les attributs sont conditionnellement indépendants donné la valeur de la classe. Cette méthode est populaire en apprentissage automatique en raison de sa simplicité, de son intuitivité, de sa rapidité et de sa capacité à gérer les attributs manquants. Les recherches montrent qu'elle fonctionne toujours bien même lorsque de fortes dépendances entre les attributs existent [22].

1.8.5 Les Réseaux de neurones artificiels

Les réseaux de neurones artificiels (RNA) sont un type d'algorithme d'apprentissage profond qui s'inspire des réseaux neuronaux biologiques. Le RNA se compose de plusieurs couches, notamment une couche d'entrée, plusieurs couches cachées et une couche de sortie, qui sont interconnectées via des nœuds. Pour fonctionner, le RNA nécessite des données d'entraînement et une sortie souhaitée, telle que la classification précise des étiquettes de groupe. L'algorithme est auto-apprenant et ne nécessite pas de fonction mathématique en entrée. Chaque couche du réseau traite les données d'entraînement et les transmet à la couche suivante, augmentant la complexité et le niveau de détail du processus d'apprentissage jusqu'à ce que la sortie souhaitée soit atteinte. Les exemples de test sont présentés au réseau appris, qui classe ensuite ces exemples. Les poids de chaque caractéristique donnent une mesure de leur importance dans le réseau [23].

Certaines limites des réseaux neurones :

- Plus de nœuds dans un réseau neuronal signifie plus de paramètres.
- Plus de paramètres peut entraîner un sur-ajustement des données.
- Plus de nœuds entraîne des coûts de calcul élevés.
- Une grande complexité rend difficile la compréhension pour l'utilisateur moyen.

1.9 Applications de classification de textes

La classification de textes est largement utilisée dans différents domaines de l'exploration de texte. Parmi les exemples courants d'utilisation de la classification de texte, on peut citer :

Filtrage et organisation de l'actualité : Aujourd'hui, la plupart des services d'actualités sont électroniques, ce qui implique la création d'un grand nombre d'articles de presse chaque jour par les organisations. Cependant, il est difficile de les organiser manuellement, d'où l'intérêt des méthodes automatisées pour la catégorisation des actualités sur une variété de portails web.

Organisation et récupération de documents : Les méthodes d'organisation et de récupération de documents sont largement applicables dans différents domaines. Les méthodes supervisées peuvent être utilisées pour organiser des documents dans des domaines tels que les bibliothèques numériques, les collections web, la littérature scientifique ou les flux sociaux. Les collections de documents organisées hiérarchiquement sont particulièrement utiles pour la navigation et la récupération.

Analyse des opinions : L'analyse des opinions consiste à exploiter les avis ou opinions des clients qui sont souvent de courts documents textuels afin de déterminer des informations utiles à partir de ces avis.

La classification des e-mails : consiste à classifier automatiquement les e-mails afin de déterminer leur sujet ou leur caractère indésirable, également appelé le filtrage des spams [12].

1.10 Les difficultés de classification de textes

Le processus de catégorisation de textes peut être entravé par de nombreuses difficultés, certaines étant courantes en apprentissage automatique supervisé, comme le sur-apprentissage. D'autres difficultés sont spécifiques aux données textuelles, telles que la

polysémie, la redondance, les variations morphologiques et l'homographie. Dans les paragraphes suivants, nous passerons en revue les trois principales difficultés liées à la catégorisation de textes.

Le sur-apprentissage : il se produit lorsque le modèle de prédiction ne parvient pas à classifier correctement de nouveaux textes, même s'il a correctement classé les textes de la base d'apprentissage. Pour éviter le sur-apprentissage, il est recommandé de sélectionner les termes les plus pertinents pour réduire la dimensionnalité. Il est suggéré d'utiliser au moins 50 à 100 fois plus de textes que de termes pour limiter le sur-apprentissage. Cependant, étant donné que le nombre de textes d'apprentissage est souvent limité, il est essentiel de réduire le nombre de termes utilisés pour éviter le sur-apprentissage, sans compromettre la pertinence du système en supprimant des termes utiles.

L'homographie : est considérer comme une ambiguïté supplémentaire, est un phénomène linguistique où deux mots s'écrivent de la même manière, mais ont des significations différentes et peuvent être prononcés différemment. Par exemple, le mot "avocat" peut désigner soit un fruit, soit un professionnel du droit.

Les mots composés : tels que "Arc-en-ciel", "peut-être", etc., sont très fréquents dans toutes les langues. Cependant, leur non-prise en charge peut considérablement réduire la performance d'un système de classification. Par exemple, si le mot "Arc-en-ciel" est traité comme trois termes séparés, cela peut conduire à des résultats incorrects. Pour résoudre ce problème, la technique des n-grammes peut être utilisée pour le codage des textes. Cette technique permet de prendre en compte les mots composés en les traitants comme des unités distinctes [24].

1.11 Conclusion

En conclusion, la classification de texte est une tâche complexe qui nécessite une compréhension approfondie des données et des algorithmes utilisés. Il existe plusieurs types de classification de texte, chacun adapté à des cas d'utilisation spécifiques. Les algorithmes de classification de texte les plus couramment utilisés sont les classificateurs naïfs de Bayes, les machines à vecteurs de support et les réseaux de neurones. Les métriques d'évaluation telles que la précision, accuracy et le F1-score sont utilisées pour évaluer les performances du modèle. Enfin, les méthodes de représentation de texte telles que le sac de mots et le modèle de langage sont utilisées pour prétraiter les données textuelles avant de les utiliser pour la

classification. La classification de texte est une technologie importante qui est largement utilisée dans de nombreuses applications pratiques et continuera à être un domaine de recherche important dans l'avenir.

Chapitre 2

Sélection des features.

2.1 Introduction

La sélection des features joue un rôle très important dans la classification de texte. Les features, également appelées variables ou caractéristiques, sont les données d'entrée utilisées pour entraîner un modèle. Son principe est de choisir les features les plus pertinentes pour le modèle afin d'améliorer la précision et la performance tout en minimisant la durée et la complexité de l'entraînement. Dans ce domaine, la sélection des features est donc essentielle pour obtenir des résultats optimaux.

Dans ce chapitre, nous allons montrer comment le processus de sélection des features fonctionne, les objectifs principaux ainsi que les métriques et les approches les plus utilisées dans le domaine de la classification du texte.

2.2 Définition

La sélection de caractéristiques en apprentissage automatique consiste à choisir un sous-ensemble de caractéristiques utilisées pour représenter les données. Cela peut être une partie de la préparation des données ou intégré dans l'algorithme d'apprentissage. L'objectif est de réduire la dimensionnalité de l'espace de caractéristiques d'origine. La sélection de caractéristiques pour l'exploration de texte est traitée séparément en raison de la spécificité des données textuelles [25].

2.3 Processus de sélection d'attributs

Le processus de recherche d'un sous-ensemble de caractéristiques consiste en quatre étapes de base :

1) la génération de sous-ensembles.

2) l'évaluation des sous-ensembles.

3) un critère d'arrêt.

4) la validation des résultats.

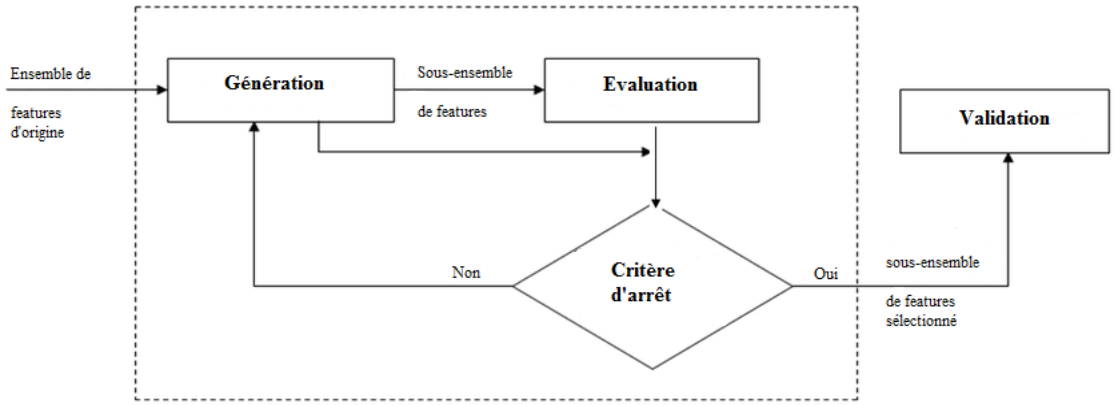

Figure 2.1-Présentation du processus de sélection des attributs [58].

La génération de sous-ensembles dépend de la stratégie de recherche dans l'espace d'états, et une fois qu'un sous-ensemble candidat est sélectionné, il est évalué à l'aide d'un critère d'évaluation. Les étapes 1 et 2 sont répétées plusieurs fois en fonction du critère d'arrêt, et le meilleur sous-ensemble de caractéristiques est sélectionné. Ce sous-ensemble est ensuite validé sur un ensemble de données indépendant ou en utilisant des connaissances de domaine, en fonction du type de tâche à accomplir [26].

2.4 Objectifs de la sélection des features

Les objectifs de la sélection des features sont les suivants :

- Réduire la dimensionnalité de l'espace de caractéristiques pour économiser les ressources et augmenter la vitesse des algorithmes.
- Éliminer les données redondantes, non pertinentes ou bruyantes.
- en améliorant la qualité des données et l'accuracy du modèle résultant, cela peut conduire à une amélioration des performances globales du modèle
- Comprendre les données pour obtenir des connaissances sur le processus qui a généré les données ou simplement pour visualiser les données [27].

2.5 Métriques de sélection pour la classification de textes

De nombreuses métriques de sélection de features ont été proposées et appliquées, dans cette section, nous en présenterons certaines :

2.5. Gain d'information (IG)

Information Gain (IG) est une mesure utilisée pour sélectionner les caractéristiques importantes dans les documents textuels en calculant la différence d'entropie de l'information entre les mots caractéristiques qui apparaissent ou non. Un score d'information gain élevé indique que le mot caractéristique transporte plus d'informations et la condition dans laquelle le score d'information gain serait à son maximum est lorsque le document appartient à une classe respective et que le terme est présent dans le document [28]. Cette mesure ne produit qu'un seul score global pour chaque mot caractéristique.

$$IG(t) = -\sum_{i=1}^{|c|} P(c_i) log P(c_i) + P(t) \sum_{i=1}^{|c|} P(c_i|t) log P(c_i|t) + P(\overline{t}) \sum_{i=1}^{|c|} P(c_i|\bar{t}) log P(c_i|\bar{t}) \quad (2.1)$$

Où $|c|$ est le nombre de classes, $P(c_i)$ est la probabilité de la classe c_i, $P(t)$ et $P(\overline{t})$ sont les probabilités de présence et d'absence du mot caractéristique t, $P(c_i|t)$ et $P(c_i|\bar{t})$ sont les probabilités conditionnelles de la classe c_i donnée la présence et l'absence du mot caractéristique t, respectivement [29].

2.5.2 Indice de Gini (GI)

GI est une méthode de sélection de caractéristiques qui mesure la pureté des caractéristiques par rapport à la classe. La pureté fait référence au niveau de discrimination d'une caractéristique pour distinguer les classes possibles. Pour une caractéristique, l'indice de Gini est calculé selon la formule :

$$GI(t_i) = \sum_{j=1}^{m} p(t_i|c_j)^2 p(c_j|t_i)^2 \quad (2.2)$$

Où m est le nombre de classes, est $p(t_i|c_j)$la probabilité d'un terme t_i étant donné une classe c_j, et $p(c_j|t_i)$est la probabilité d'une classe étant donné le terme t_i [30].

2. 5.3 Chi-Square

Le test du chi-deux (anglais : Chi-Square) est utilisé pour évaluer deux types de comparaisons : les tests d'indépendance et les tests d'ajustement à une loi de probabilité [54], et est une méthode statistique utilisée pour mesurer la différence entre les fréquences attendues et les fréquences observées pour deux événements. En ce qui concerne la sélection de fonctionnalités, les deux événements concernent l'occurrence d'un terme et l'occurrence d'une classe. La valeur de chaque terme par rapport à la valeur de la classe est calculée en utilisant l'équation suivante :

$$X^2(t,c) = \frac{N(AD-BC)^2}{(A+C)(B+C)(A+B)(C+D)} \quad (2.3)$$

Où N est le nombre total de documents, A est le nombre d'occurrences de t et c, B est le nombre d'occurrences de t sans c, C est le nombre d'occurrences de c sans t, et D est le nombre de non-occurrences de t et c [30].

2.5.4 Fréquence des documents

La fréquence de document (DF) est une mesure qui détermine le nombre total de documents dans une collection qui contiennent un terme donné. Elle est utilisée pour sélectionner les termes les plus informatifs pour la classification. Les termes rares sont considérés comme non-informatifs et sont supprimés. Cependant, la DF peut avoir des limitations, car certains termes fréquents peuvent ne pas être discriminants, tandis que certains termes rares peuvent être importants. Malgré cela, la DF est une méthode simple et efficace de sélection de fonctionnalités avec une complexité temporelle linéaire par rapport au nombre de documents [31].

2.5.5 Information Mutuelle (MI)

MI est une mesure fréquemment utilisée en théorie de l'information pour mesurer la dépendance mutuelle entre deux variables. Plus précisément, la valeur de MI entre un terme s_i et une étiquette de classe c_j est définie comme suit :

$$MI(s_i, c_j) = log \frac{p(s_i,c_j)}{p(s_i)p(c_j)} \quad (2.4)$$

Le biais en faveur des termes rares et la sensibilité aux erreurs d'estimation de probabilité sont deux limites de l'information mutuelle [31].

2.6 Approches de sélection des features

Les méthodes de sélection de caractéristiques peuvent être classées de plusieurs manières. La plus courante est la classification en filtres (en.filter), enveloppes (en.wrapper), intégrées (en.embedded) et méthodes hybrides(en. Hybrid). La classification mentionnée ci-dessus suppose une indépendance ou une quasi-indépendance des caractéristiques. Des méthodes supplémentaires ont été élaborées pour les ensembles de données comportant des caractéristiques structurées dans lesquelles des dépendances existent [26].

2.6.1 Approche filtre

Les méthodes de filtrage évaluent la sélection de caractéristiques en fonction de leur classement, en utilisant des tests statistiques pour déterminer la corrélation de chaque caractéristique avec la classe. Les caractéristiques avec des scores inférieurs à un certain seuil sont supprimées, tandis que celles ayant des scores supérieurs sont sélectionnées. Les méthodes de filtrage sont indépendantes de l'algorithme de classification, ce qui les rend libres de son biais et réduit le sur-ajustement. Cependant, cette indépendance signifie également que l'interaction avec le classificateur n'est pas prise en compte lors de la sélection de caractéristiques, ce qui donne un ensemble de caractéristiques moins ajusté. Les méthodes de filtrage ont l'avantage d'être moins exigeantes en termes de calcul et peuvent être adaptées à des données de haute dimension. Cependant, elles peuvent produire des modèles avec une performance prédictive réduite par rapport aux méthodes enveloppes ou intégrées [32].

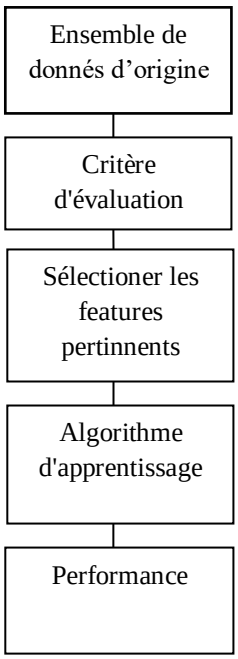

Figure 2.2- Principe de l'approche filter [59].

2.6.2 Approche enveloppe

Les méthodes enveloppes reposent sur un algorithme d'apprentissage prédéfini pour évaluer la qualité des features sélectionnés. Le processus implique deux étapes : la recherche de l'ensemble de features et l'évaluation de ces derniers. Le composant de recherche de features génère un sous-ensemble de features, qui est évalué par l'algorithme d'apprentissage pour

déterminer sa qualité en fonction de la performance d'apprentissage. Le processus se répète jusqu'à ce que la performance d'apprentissage désirée soit atteinte ou que certains critères d'arrêt soient satisfaits. Finalement, l'ensemble de features avec la meilleure performance d'apprentissage est sélectionné et renvoyé en sortie [33].

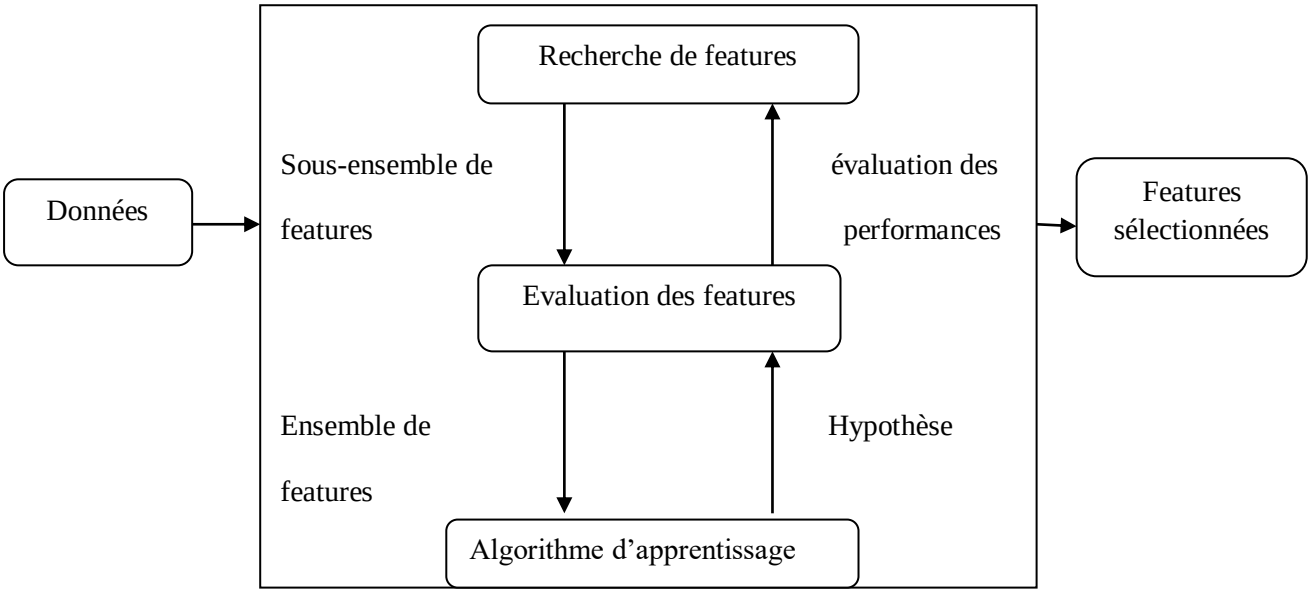

Figure 2.3- Principe de l'approche enveloppe [33].

Les inconvénients majeurs des méthodes enveloppes sont : la complexité de calcul, la durée est plus longue et le risque élevé de sur-ajustement.

Les principaux avantages des méthodes enveloppes est qu'elles interagissent avec un classificateur pour la sélection de caractéristiques et qu'elles prennent en compte les dépendances entre les caractéristiques, ce qui est un avantage par rapport aux méthodes de filtrage [34].

2.6.3 Approche intégrée

Cette technique intégrée fournit une solution de compromis entre la méthode de filtre et la méthode d'enveloppe, ce qui peut résoudre la redondance élevée de l'algorithme de filtre et la complexité de calcul de l'algorithme d'enveloppe. La sélection de caractéristiques intégrée est automatiquement effectuée pendant le processus d'apprentissage du modèle, en d'autres termes, le processus de recherche et de sélection des sous-ensembles de caractéristiques est intégré à la construction du classificateur [35].

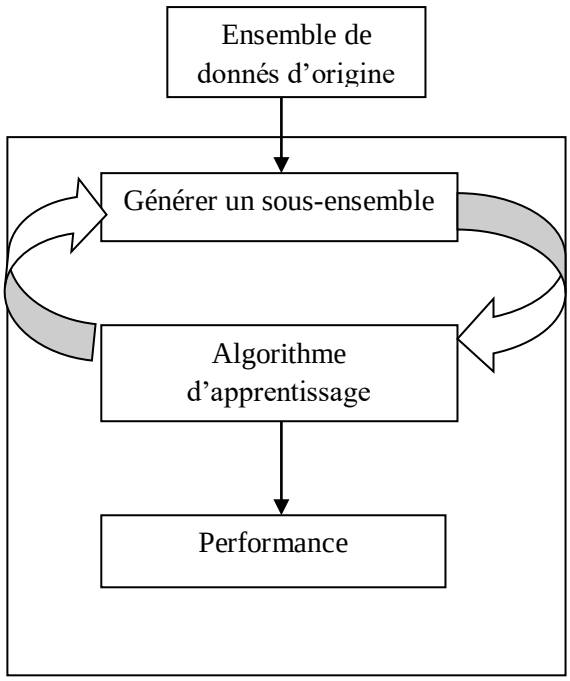

Figure 2.4- Principe de l'approche intégrée [59].

Les méthodes intégrées présentent des avantages par rapport à la méthode enveloppe, notamment une moindre intensité computationnelle, des temps d'exécution plus rapides et un risque de sur-ajustement potentiellement plus faible. Cependant, il est important de noter que la sélection d'un petit groupe de caractéristiques peut être problématique dans certains cas [34].

2.6.4 Approche hybride

Cette approche est la combinaison entre Filter et enveloppe et elle est adoptée dans l'espoir de regrouper les avantages des deux méthodes. Elle consiste à sélectionner les caractéristiques pertinentes (éliminer les caractéristiques redondantes) tout en améliorant la performance du système de reconnaissance. Cette dernière approche sera utilisée dans nos travaux. Pour les méthodes hybrides, le processus de sélection des caractéristiques est effectué conjointement au processus de classification. Une fonction d'évaluation de type "filter" est tout d'abord utilisée pour présélectionner les sous-ensembles de caractéristiques les plus discriminantes tout en fait une bonne discrimination entre les classes, Puis les taux moyens de bonne reconnaissance obtenus après sélection sont comparés avec ceux obtenues avant sélection, afin de déterminer le sous ensemble final [36].

2.7 Conclusion

En conclusion, la sélection des features est une étape essentielle pour la création de modèles de l'apprentissage automatique performants.

Dans ce chapitre, nous avons exploré la définition de la sélection de features, présenté les objectifs fondamentaux de cette étape et montré les métriques les plus utilisées dans la littérature telles que IG, MI, CH2, etc. Nous avons également présenté différentes approches telles que le filtrage, enveloppe, intégrée et hybride, en détaillant leurs étapes de fonctionnement.

Chapitre 3

Les algorithmes génétiques

3.1 Introduction

Les algorithmes génétiques sont une classe d'algorithmes d'optimisation inspirés du processus de sélection naturelle dans l'évolution biologique. Ils constituent une méthode puissante et polyvalente pour résoudre des problèmes d'optimisation complexes dans de nombreux domaines.

Dans ce chapitre, nous allons explorer les fondamentaux des algorithmes génétiques, y compris leurs principes de base, les avantages et les inconvénients de leur utilisation, et quelques exemples pratiques de leurs applications.

3.2 L'évolution des algorithmes génétiques

L'informatique évolutive, introduite pour la première fois par I. Rechenberg dans les années 1960 et développée par d'autres chercheurs par la suite, repose sur la théorie de l'évolution présentée par Charles Darwin en 1859 pour expliquer ses observations des plantes et des animaux dans l'écosystème naturel.

La théorie stipule que dans chaque nouvelle génération, les individus moins performants ont tendance à perdre la bataille pour la survie dans la compétition pour la nourriture. Les algorithmes génétiques (GAs) ont été inventés et développés par John Holland en 1975 en tant que méthode heuristique basée sur la "survie du plus apte" [37].

3.3 Définition

Un algorithme génétique est un type d'algorithme évolutionnaire qui utilise l'exploration et l'exploitation pour trouver des solutions exactes ou approximatives à des problèmes d'optimisation et de recherche.

Dans le processus des algorithmes génétiques, les stratégies d'exploration et d'exploitation sont toutes deux utilisées. La phase d'exploration consiste à découvrir de nouvelles solutions à l'intérieur de l'espace des solutions, tandis que la phase d'exploitation utilise les meilleures solutions trouvées lors de recherches précédentes.

Le GA combine ces idées en commençant par une population de chromosomes initiaux qui représentent des solutions, qui peuvent être décrits par des chaînes de bits ou des expressions symboliques. La fitness de chaque chromosome est évaluée, et les meilleurs individus sont sélectionnés de manière probabiliste pour la génération suivante en utilisant des techniques inspirées de la biologie évolutive, telles que l'héritage, la mutation, la sélection et le

croisement. L'algorithme simule la sélection naturelle, avec les individus moins adaptés étant remplacés par de meilleures solutions découvertes grâce à l'exploration et l'exploitation. La recherche de chromosomes appropriés se poursuit jusqu'à ce qu'une condition d'arrêt soit atteinte [38,39].

3.4 Espace de recherche

L'espace de recherche ou l'espace d'état fait référence à l'ensemble de toutes les solutions réalisables pour un problème donné. Chaque point dans cet espace représente une solution réalisable, et notre objectif est de trouver la meilleure solution parmi celles-ci. La fitness ou la valeur de chaque solution réalisable permet d'évaluer sa pertinence pour le problème. La recherche d'une solution consiste à trouver un extrême, qu'il s'agisse d'un minimum ou d'un maximum, dans cet espace de recherche. Cependant, en raison de la complexité de l'espace de recherche, il peut être difficile de déterminer par où commencer la recherche [40].

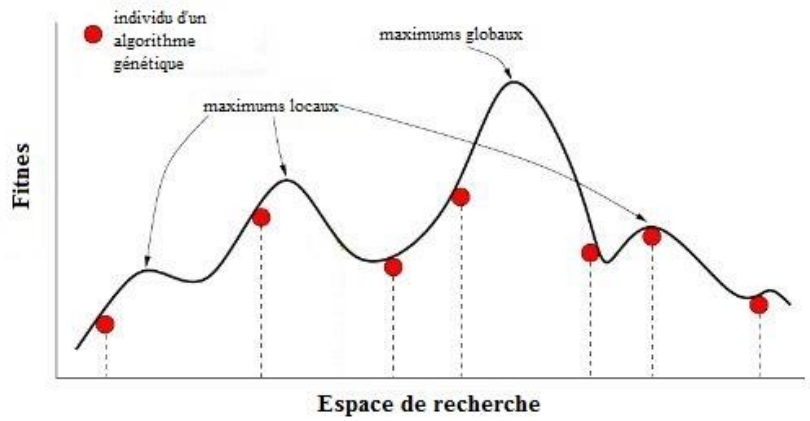

Figure 3.1. Exemple d'un paysage de fitness [41].

La Figure 3.1 illustre un paysage de fitness hypothétique qui met en évidence le fonctionnement d'un algorithme génétique. La valeur de fitness de chaque individu dans la population est déterminée par sa position sur le paysage. Dans un GA, les individus de la population peuvent se trouver dans différentes régions du paysage. Le GA bénéficie de l'accès à l'information globale grâce à la population, ce qui lui permet d'explorer diverses régions du paysage. En utilisant des opérateurs génétiques, le GA peut guider la population vers des zones plus prometteuses, facilitant ainsi la recherche de solutions optimales [41].

3.5 Terminologie

Population : représente un sous-ensemble de toutes les solutions possibles pour résoudre un problème spécifique.

Chromosome : appelé également un individu, un chromosome représente l'une des solutions possibles pour un problème spécifique.

Gène : est un élément constitutif d'un chromosome, occupant une position spécifique. Chaque gène possède deux propriétés :

Allele : il s'agit de la valeur qu'un gène adopte dans un chromosome spécifique.

Locus: il s'agit de la position spécifique d'un gène sur un chromosome.

Chaque chromosome est représenté de deux manières différentes :

Génotype: c'est l'ensemble des gènes qui représentent le chromosome.

Phénotype: c'est la représentation physique et réelle du chromosome [42,43].

3.6 Les composants

Dans cette section, nous présentons les éléments clés pour les algorithmes génétiques :

1- **Le codage :** Cette étape consiste à déterminer comment représenter la solution d'un problème sous forme de chromosomes ou d'ensembles de chromosomes. Deux types couramment utilisés sont le codage binaire et le codage réel. La qualité du codage des données conditionne le succès de l'algorithme génétique.

2- **Initialisation :** Cela implique la génération d'une population initiale de chromosomes. L'initialisation peut se faire de manière aléatoire si aucune information spécifique n'est disponible, sinon elle peut être réalisée à l'aide d'heuristiques appropriées. Le choix de la méthode d'initialisation dépend du problème.

3- **La fonction de fitness :** Chaque individu de la population se voit attribuer une valeur de fitness en fonction de sa performance dans le domaine du problème. Cette mesure de fitness évalue à quel point chaque individu est adapté à la résolution du problème.

4- **Les opérateurs :** tels que l'opérateur de sélection pour choisir les meilleurs individus pour la reproduction, l'opérateur de croisement pour créer de nouveaux individus en combinant des parties de deux parents et l'opérateur de mutation pour introduire des changements aléatoires dans les chromosomes.

   Une explication détaillée de chaque composant sera fournie dans la section 8.

5- **Les paramètres :** tel que la taille de la population, nombre total de générations, probabilités des opérateurs de croisement et de mutation.
   Une explication détaillée de chaque composant sera fournie dans la section 9.
6- **Critère d'arrêt :** L'algorithme arrête après ces critères : Atteinte du nombre maximum de générations, expiration d'un temps spécifié, absence de changement de la performance optimale pendant un certain nombre de générations [1]. Lorsque la population ne présente plus d'évolution ou évolue insuffisamment, cela signifie qu'elle est devenue homogène et que l'on peut supposer qu'elle se trouve près de l'optimum [44].

3.7 Le fonctionnement d'algorithme génétique

La figure ci-dessous est un schéma illustrant les fonctions de base d'un algorithme génétique.

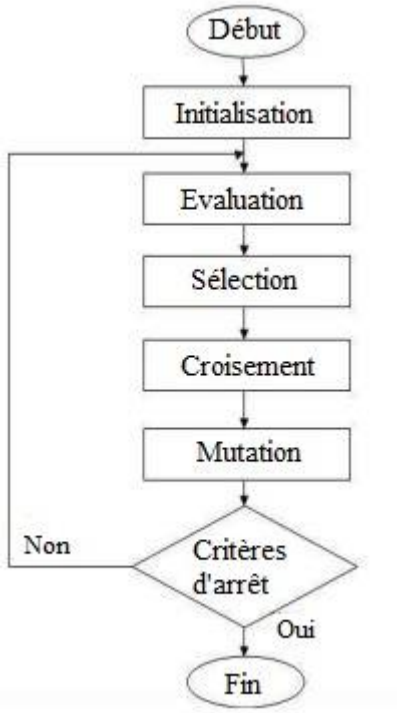

Figure 3.2- Le diagramme génétique [60].

Le processus de l'algorithme génétique commence par la génération aléatoire d'un ensemble de chromosomes, qui forme la population initiale. La valeur de la fitness de chaque chromosome dans la population est ensuite évaluée, et le plus optimal est retenu. De nouveaux descendants (offspring) sont créés à l'aide d'opérations génétiques, notamment la

sélection, le croisement et la mutation. Ce processus est répété plusieurs fois jusqu'à la terminaison de l'algorithme [39].

3.8 Variantes des GAs

Les algorithmes génétiques sont une famille d'algorithmes fondés sur des principes communs, mais qui peuvent varier selon la représentation choisie et les opérateurs de croisement, de mutation et de sélection utilisés. Dans cette section, les choix les plus fréquemment adoptés pour définir les différentes variantes des algorithmes génétiques seront présentés [45].

3.8.1 Le codage

La première étape de l'application d'un algorithme génétique est la représentation des différentes valeurs possibles de la variable pour laquelle on cherche la valeur optimale. Cette représentation est réalisée en utilisant un code adapté qui permettra d'établir un lien entre les valeurs de la variable et les individus de la population. Le choix du code à utiliser dépend du type de problème traité et peut avoir un impact significatif sur la performance globale de l'algorithme [46]. Deux types de codage ont été identifiés dans la littérature : binaire et réelle.

3.8.1.1 Codage binaire

Le codage classique repose sur l'utilisation de l'alphabet binaire 0 et 1 pour représenter les chromosomes. Ce codage ne nécessite aucune spécification pour les opérateurs génétiques, car toute manipulation d'un chromosome donne un nouveau chromosome valide. Toutefois, le codage binaire peut poser des difficultés pratiques car il peut être difficile ou lourd de coder certaines solutions de cette manière. De plus, dans certains cas, la taille de la mémoire nécessaire peut devenir trop grande [45].

3.8.1.2 Codage réelle

L'utilisation du codage réel peut être plus pratique que le codage binaire pour certains problèmes d'optimisation. Le codage réel utilise des nombres réels pour représenter les gènes des chromosomes, évitant ainsi la conversion en code binaire. Cela permet d'améliorer l'efficacité de l'algorithme génétique en évitant les opérations de décodage supplémentaires. De plus le codage réel permet de réduire la longueur des chromosomes en comparaison avec le codage binaire [45].

3.8.2 La fonction de fitness

L'opérateur d'évaluation est crucial dans l'algorithme génétique car il permet de sélectionner les individus les plus performants pour poursuivre l'optimisation. Il utilise une fonction d'évaluation qui attribue un poids à chaque individu appelé "fitness". Ensuite, l'opérateur de sélection utilise les fitness pour calculer la force de chaque chromosome dans la population et sélectionner les plus forts (sélection). Ces chromosomes sont ensuite modifiés par les opérateurs de croisement et mutation. Cette fonction peut être complexe et dépend des contraintes du problème à résoudre.

Ces deux éléments sont spécifiques à chaque problème et doivent être soigneusement définis avant l'application de l'algorithme génétique. Une fois que le codage et l'évaluation sont déterminés, l'algorithme génétique utilisé pour résoudre le problème sera toujours le même [45].

3.8.3 Initialisation

Dans le domaine de l'optimisation, il est important de trouver l'optimum aussi rapidement que possible. Pour y parvenir, il faut disposer de points de départ de haute qualité. Dans les cas où la position de l'optimum est inconnue, il est courant de créer des individus de manière aléatoire, tout en respectant les contraintes et uniformément dans chaque domaine. Toutefois, si l'on dispose d'informations préalables sur le problème, il est préférable de générer des individus dans des zones spécifiques afin d'optimiser la convergence de l'algorithme génétique [44].

3.8.4 La sélection

L'opérateur de sélection joue un rôle crucial en permettant de sélectionner les individus les plus adaptés pour contribuer à la génération suivante. Ce processus de sélection se base sur la valeur de la fonction d'adaptation. Les meilleurs individus sont copiés pour former la nouvelle population.

Ce processus est inspiré de la sélection naturelle de Darwin (survie du plus apte), il simule cette sélection artificiellement [45]. Plusieurs types de sélection ont été identifiés dans la littérature :

3.8.4.1 La sélection par classement

Elle consiste à la sélection des individus les plus performants dans une population, en les classant dans un ordre croissant ou décroissant en fonction de leur score d'adaptation. Seuls les meilleurs individus sont conservés, mais cela peut conduire à une convergence prématurée de l'algorithme génétique.

Il est important de maintenir une certaine diversité dans la population des algorithmes génétiques en conservant des individus moins performants. Cependant, cela peut poser un problème si une limite de sélection est fixée, car cela peut empêcher de conserver des candidats prometteurs pour les générations futures [45].

3.8.4.2 La sélection par la roulette

Elle implique la création d'une roue de loterie biaisée où chaque individu de la population est représenté par une section proportionnelle à sa valeur de fitness. Cela permet même aux individus les plus faibles d'avoir une chance de survie [45].

Pour effectuer la sélection par la méthode de la roulette :

- Additionnez les valeurs de fitness de tous les individus de la population.
- Calculez la probabilité de chaque individu d'être sélectionné en divisant sa fitness par la somme totale de fitness de la population.
- Divisez la roue de la roulette en sections en fonction des probabilités calculées à l'étape précédente.
- Faites tourner la roue un certain nombre de fois (n) et sélectionnez à chaque fois l'individu dans la section où se trouve le curseur.

La probabilité de sélection d'un individu a_j :

$$Ps(a_i) = \frac{f(a_i)}{\sum_{i=1}^{n}(a_j)}; j = 1,2,3,\dots,n \qquad (3.1)$$

Où n est la taille de la population, $f(a_i)$ est la valeur de fitness de l'individu a_i [47].

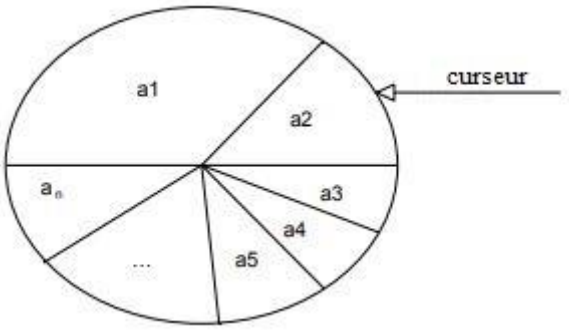

Figure 3.3- La sélection par la roulette [47].

3.8.4.3 La sélection par tournoi

C'est la sélection dans laquelle 'n' individus sont choisis au hasard dans la population et se disputent entre eux. L'individu ayant la valeur de fitness la plus élevée remporte le tournoi et est sélectionné pour un traitement ultérieur. La taille du tournoi, fait référence au nombre d'individus.

Cette méthode présente plusieurs avantages par rapport aux autres techniques de sélection, notamment une complexité temporelle réduite. Le risque de domination par des individus forts est réduit et pas de nécessité de trier les valeurs de fitness. Cependant, plus la taille du tournoi est grande, plus la probabilité de perte de diversité est élevée en raison de la sélection aléatoire ou de la perte dans les populations intermédiaires [47].

3.8.4.4 L'élitisme

Consiste à préserver le meilleur individu d'une population en lui permettant de participer à la reproduction pour transmettre ses caractéristiques avantageuses aux générations suivantes. Cela permet d'éviter la perte des meilleurs individus dus aux opérations de croisement et de mutation [48].

3.8.5 Le croisement

Le croisement est une étape cruciale pour explorer l'espace des solutions envisageables. Son principe consiste à combiner aléatoirement une partie des gènes de chacun des deux parents afin de créer un nouvel individu. Une fois la sélection des individus terminée, ils sont répartis aléatoirement en couples. Les chromosomes des parents sont copiés et recombinés pour produire deux descendants ayant des caractéristiques de leurs deux parents [45].

L'opérateur de croisement utilise une probabilité de croisement p_c.

Plusieurs types de sélection ont été identifiés dans la littérature :

3.8.5.1 Croisement en 1-point

Son principe est simple, pour chaque couple de chromosomes un seul point de croisement est choisi au hasard, divisant chaque chromosome en deux sections. Les sections après le point de coupe sont ensuite échangées pour créer deux nouveaux chromosomes descendants (offspring).

Parent1 :	**0**	**1**	**1**	**0**	**1**	**1**	**0**	**1**
Parent2 :	1	1	0	0	1	0	0	1
Fils 1 :	**0**	**1**	**1**	**0**	**1**	0	0	1
Fils 2 :	1	1	0	0	1	**1**	**0**	**1**

Figure 3.4- Croisement en 1-point [45].

3.8.5.2 Croisement en 2-points

C'est un cas particulier du croisement en n-points. On choisit aléatoirement deux points de coupure pour créer les descendants [45].

3.8.5.3 Croisement en n-points

Il existe deux types de méthodes de croisement qui diffèrent par le nombre de points de croisement utilisés. Lorsqu'un nombre pair de points de croisement est utilisé, ils sont sélectionnés au hasard autour d'un cercle, et les informations génétiques sont échangées. Cependant, lorsque le nombre de points de croisement est impair, un point de croisement différent est toujours présumé au début de la chaîne [37].

3.8.5.4 Croisement uniforme

Ce type de croisement utilise un masque de croisement binaire de même longueur que les chromosomes, généré aléatoirement. Le masque détermine lequel des parents fournit le gène correspondant pour chaque gène de la descendance. Dans les cas où le masque a un 1 à une position particulière, le gène est copié du premier parent, et s'il a un 0, le gène est copié du

deuxième parent. Pour chaque paire de parents, un nouveau masque de croisement est produit, ce qui donne une descendance avec un mélange de gènes provenant de chaque parent [37].

Parent 1	1 0 1 1 0 0 1 1
Parent 2	0 0 0 1 1 0 1 0
Mask	1 1 0 1 0 1 1 0
Fils 1	1 0 0 1 1 0 1 0
Fils 2	0 0 1 1 0 0 1 1

Figure 3.5- Croisement uniforme [37].

3.8.5.5 Croisement réel

Le codage réel requiert des opérateurs génétiques spécifiques pour la manipulation des chromosomes. Il est de plusieurs types [45] :

a) Ordre de base cyclique

Pour créer un fils, il suffit de copier une sous-chaîne d'un parent et de compléter les gènes manquants à partir de l'autre parent, en maintenant l'ordre des gènes. Généralement, une fois deux chromosomes parents sélectionnés pour le croisement, deux points de coupures sont choisis aléatoirement sur chaque parent. Ensuite on place les sous-chaînes entre les points de coupure sur les deux fils dans la même position que les parents. Pour compléter les gènes manquants du fils 1, on commence par insérer les gènes situés à droite du deuxième point de coupure du parent 2 tout en gardant l'ordre des gènes et en ignorant les gènes déjà pris. Le deuxième fils est complété à partir du parent 1 de la même manière que le fils 1 [45].

			1er pt				2ème pt		
Père 1	a	b	c	d	e	f	g	h	i
Père 2	f	b	g	a	e	i	c	h	d
Fils 1	.	.	.	d	e	f	g	.	.
Fils 2	.	.	.	a	e	i	c	.	.
Fils 1	a	i	c	d	e	f	g	h	b
Fils 2	d	f	g	a	e	i	c	h	b

Figure 3.6- Croisement d'ordre de base cyclique [45].

b) Croisement d'ordre maximal

Ce type de croisement a pour objectif de garder le maximum possible les positions et l'ordre des gènes. On commence par choisir aléatoirement deux points de coupure. Les sous-chaînes situées au milieu sont inter-changées. Les gènes manquants sont par la suite complétés à partir de chaque père en allant de gauche à droite et en choisissant le premier caractère disponible. A la différence du croisement de base cyclique, le fils 1 est complété à partir du parent 1 et le fils 2 à partir du parent 2 [45].

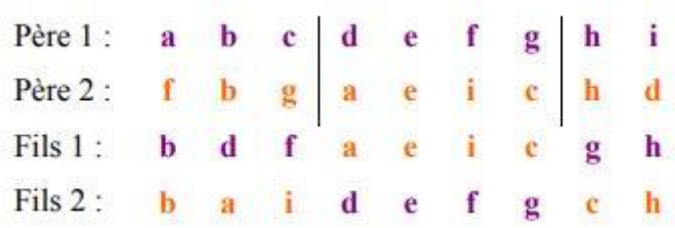

Figure 3.7- Croisement d'ordre maximal [45].

3.8.6 Mutation

La mutation consiste en la modification aléatoire d'une partie d'un chromosome, ce qui permet une exploration aléatoire de l'espace des séquences génétiques [45].

La mutation empêche l'algorithme de rester bloqué dans les optima locaux et favorise la diversité génétique au sein de la population [37].

L'opérateur de mutation utilise une probabilité de mutation p_m.

3.8.6.1 Mutation en codage binaire

Dans un algorithme génétique simple, la mutation en codage binaire est la modification aléatoire occasionnelle (de faible probabilité) de la valeur d'un caractère de la chaîne [45].

3.8.6.2 Mutation en codage réel

Pour le codage réel, les opérateurs de mutation les plus connus et les plus utilisés sont les suivants [45] :

a) L'opérateur d'inversion simple

Consiste à choisir aléatoirement deux points de coupure et inverser les positions des bits situés au milieu [45].

Exemple illustratif :

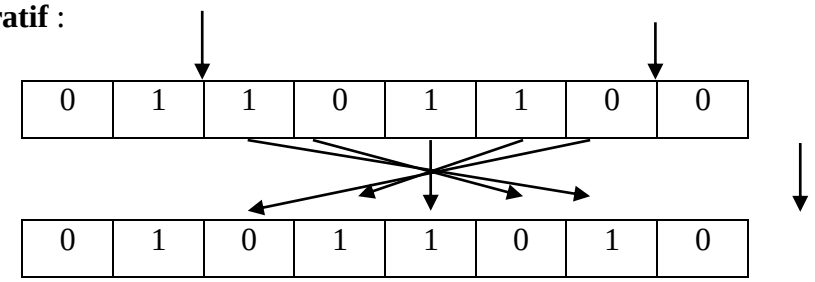

Figure 3.8- Mutation par inversion simple.

Supposons que nous choisissions les positions 2 et 8 comme points de coupure. Nous inverserons alors les bits entre ces deux positions en d'autres termes les bits aux positions 3, 4, 5, 6 et 7 (en comptant de gauche à droite) ont été inversés.

b) L'opérateur d'insertion

Consiste à sélectionner au hasard un bit et une position dans le chromosome à muter, puis à insérer le bit en question dans la position choisie [45].

Exemple illustratif :

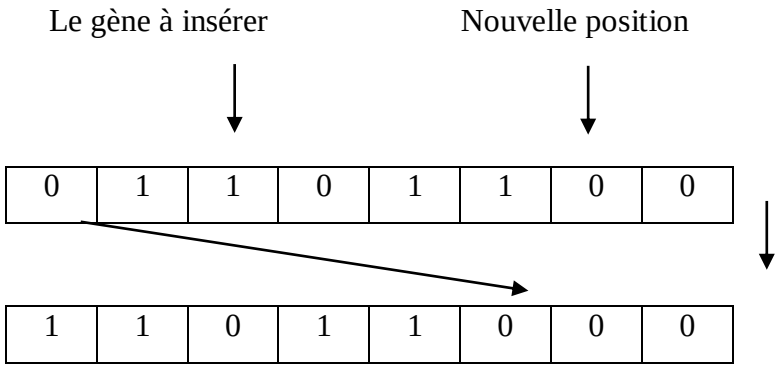

Figure 3.9- Mutation par insertion.

Supposons que nous choisissions le premier gène (0) à insérer à la position 6 du chromosome. Pour cela, nous devons décaler tous les gènes existants à partir de la position 6 vers la droite pour faire de la place au gène à insérer.

c) L'opérateur d'échange réciproque

Cet opérateur permet la sélection de deux bits et les inter changés [45].

Exemple illustratif :

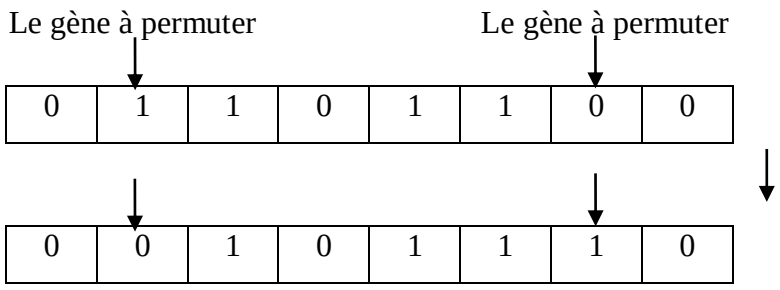

Figure 3.10- mutation par échange réciproque.

Si nous choisissons les positions 2 et 7 pour l'opérateur d'échange réciproque, nous allons permuter les gènes situés à ces deux positions. C'est-à-dire le gène à la position 2 devient 0 et le gène à la position 7 devient 1.

3.9 Valeurs des paramètres

Ces paramètres conditionnent la convergence d'un algorithme génétique [45] :

Si la population est trop importante, cela entraîne une augmentation du temps de calcul et une demande d'espace mémoire considérable. En revanche, une population trop réduite peut conduire à la convergence prématurée vers un optimum local [45].

La probabilité de croisement est un paramètre qui indique à quelle fréquence le croisement des chromosomes des parents aura lieu lors de la création de la progéniture. Si cette opération n'a pas lieu, les descendants seront des copies exactes des parents. En revanche, si le croisement a lieu, la progéniture sera issue de parties des chromosomes des deux parents. Si la probabilité de croisement est de 100%, tous les descendants seront créés de cette manière, alors que si elle est de 0%, une nouvelle génération sera créée à partir de copies exactes des chromosomes de la population précédente.

Une probabilité relativement plus élevée pc pour le croisement est utilisé dans l'intervalle de 0,6 à 0,95 [37,49].

La probabilité de mutation (Pm) dans la technique de mutation détermine la fréquence à laquelle les parties du chromosome seront mutées. Si aucune mutation n'a lieu, la progéniture est générée immédiatement après le croisement (ou directement copiée) sans aucun changement. En revanche, si une mutation est effectuée, une ou plusieurs parties du chromosome sont modifiées. Si la probabilité de mutation est de 100%, le chromosome entier est modifié, tandis que si elle est de 0%, rien ne sera changé. La mutation ne doit pas être trop fréquente, car cela transformerait l'algorithme génétique en une simple recherche aléatoire.

Une probabilité pm utilisé est généralement très faible dans l'intervalle de 0,001 à 0,05 [37,49].

3.10 Les avantages et les limites

L'algorithme génétique possède ces avantages :

Les algorithmes génétiques sont efficaces pour résoudre des problèmes avec des paysages de fitness complexes. Ces paysages se caractérisent par des fonctions de fitness discontinues, qui évoluent avec le temps ou qui comportent plusieurs optima locaux. La plupart des problèmes offrent une large gamme de solutions potentielles [50].

L'algorithme génétique se distingue par sa rapidité et son efficacité supérieures par rapport aux autres algorithmes traditionnels [51].

Son résultat final consiste à présenter une liste de solutions ou de solutions considérées comme étant de qualité, plutôt que de se limiter à une seule solution [51].

Finalement, l'algorithme génétique nous fournit toujours une solution au problème donné, qui s'améliore également au fil des générations [51].

L'approche des algorithmes génétiques est particulièrement adaptée et utile pour les espaces de recherche étendus et lorsque de nombreux paramètres sont présents [51].

L'algorithme génétique possède ces limites :

Le choix des paramètres, y compris la fonction de fitness, la taille de la population et les taux de mutation et de croisement, doit être soigneusement pris en compte dans un algorithme génétique. Par exemple, si la taille de la population est trop petite, l'algorithme génétique risque de ne pas explorer suffisamment l'espace des solutions, ce qui limite sa capacité à trouver de manière cohérente de bonnes solutions [51].

En raison de la nécessité de calculer continuellement la "valeur de fitness", les algorithmes génétiques peuvent entraîner des coûts de calcul élevés [51].

Convergence prématurée : Cela fait référence à une situation où un individu qui est très adapté par rapport à ses concurrents domine rapidement la population, ce qui entraîne l'algorithme à rester piégé dans un optimum local au lieu d'explorer suffisamment l'espace de recherche pour trouver l'optimum global [50].

3.11 Domaines d'application

Les domaines d'application d'algorithme génétique [37] sont les suivants:

- Robotiques.
- La planification de tâches, systèmes de fabrication.
- Apprentissage automatique : Conception de réseaux de neurones, à la fois en termes d'architecture et de pondération, amélioration des algorithmes de classification, systèmes de classification.
- Optimisation combinatoire : voyageur de commerce (TSP), routage.
- Traitement d'images.
- Analyse de données.

3.12 Conclusion

Dans ce chapitre nous avons vu comment l'algorithme génétique fonctionne en détail et examiné les composants clés tels que les opérateurs de fitness et d'encodage. Avec une compréhension approfondie de ces éléments, nous sommes prêts à appliquer des algorithmes génétiques pour résoudre des problèmes d'optimisation complexes dans divers domaines. En combinant la puissance des algorithmes génétiques avec une connaissance spécialisée du domaine, nous pouvons découvrir des solutions optimales pour une large gamme de problèmes pratiques.

Chapitre 4

La méthode proposée.

4.1 Introduction

La sélection de caractéristiques (features) est une étape clé dans la tâche de classification, permettant de sélectionner les caractéristiques les plus informatives pour but de réduire la dimensionnalité de l'ensemble de données et d'améliorer la précision et la performance de la tâche de classification. Cependant, cette étape peut être difficile et coûteuse en termes de temps et de ressources, surtout lorsque l'ensemble de données contient un grand nombre de caractéristiques comme dans le cas des données textuelles.

Les méthodes de sélection basées sur le filtre (filter-based) sont les plus adéquates pour les données de grande dimensionnalité dû principalement à leur faible coût de calcul. Cependant, les caractéristiques sélectionnées qui ont des scores proches ou similaires impliquent des features redondants pour le processus de classification.

Dans ce chapitre, nous présentons une méthode hybride pour la sélection de caractéristiques basée sur le gain d'information (Information Gain) et les algorithmes génétiques, qui combine les avantages des deux approches mentionnées pour aboutir à un ensemble de features optimal et sans redondance pour la tâche de classification.

4.2 Impact des features redondants sur les performances et le temps d'exécution du classifieur

Les caractéristiques redondantes se réfèrent à des attributs dans un ensemble de caractéristiques qui sont fortement corrélés avec d'autres caractéristiques présentes, mais qui ne contribuent pas de manière significative à améliorer la capacité discriminatoire de l'ensemble de caractéristiques [52].

Les features redondantes dans un problème de classification peuvent avoir un impact significatif sur les performances du modèle et le temps nécessaire à son exécution :

1- L'inclusion de features redondantes augmente la dimensionnalité des données, ce qui peut entraîner une augmentation significative du temps nécessaire pour entraîner le classifieur. Plus le nombre de features est élevé, plus les calculs nécessaires deviennent complexes.
2- La présence de features redondantes peut entraîner une réduction des performances d'un classifieur. Lorsque des features redondantes sont introduites, elles peuvent

apporter des informations similaires ou non informatives pour la tâche de classification, ce qui peut entraîner une augmentation du bruit dans les données.

Le bruit supplémentaire rend plus difficile la séparation des classes, Cela peut conduire à une baisse d'accuracy du classificateur réduisant ainsi sa capacité à généraliser correctement sur de nouvelles instances.

Pour remédier à ces problèmes, des techniques de sélection de caractéristiques sont utilisées pour identifier et éliminer les caractéristiques redondantes.

4.3 La méthode proposée

L'une des principales raisons d'utiliser une approche hybride pour la sélection des caractéristiques est qu'il n'existe pas de solution directe pour sélectionner les meilleures caractéristiques et éliminer celles qui sont redondantes.

L'un des principaux inconvénients de l'approche de sélection de caractéristiques basée sur le gain d'information est qu'elle ne prend pas en compte les interactions entre les caractéristiques et le classificateur. L'importance d'une caractéristique peut changer lorsque d'autres caractéristiques sont prises en compte. De plus, l'approche de gain d'information peut sélectionner des caractéristiques redondantes et non informatives [53].

Donc le problème principal de l'IG est la redondance, qui peut causer des défis majeurs tels qu'une mauvaise performance, une augmentation de la complexité du modèle et une augmentation de la dimension des caractéristiques sélectionnées. C'est pourquoi il peut être judicieux de combiner le gain d'information avec d'autres méthodes de sélection de caractéristiques, telles que les algorithmes génétiques, pour obtenir un ensemble plus complet et optimisé de caractéristiques.

Les algorithmes génétiques sont choisis pour résoudre le problème de redondance en offrant une exploration globale de l'espace des caractéristiques et en évaluant simultanément différentes combinaisons de caractéristiques. Ils utilisent des techniques de sélection naturelle pour favoriser les caractéristiques les plus performantes, réduisant ainsi la redondance dans l'ensemble final. Les opérations de croisement et de mutation introduisent de nouvelles variations, évitant les solutions moins optimales et favorisant l'exploration de régions plus discriminantes de l'espace des caractéristiques.

En résumé, les différentes méthodes de sélection de caractéristiques ont leurs propres avantages et limites, et en les combinant dans une approche hybride, nous pouvons surmonter les limites de chaque méthode individuelle et exploiter leurs forces pour obtenir de meilleures performances et une meilleure accuracy.

La méthode que nous avons proposée suit quatre étapes principales : La première étape est le prétraitement, il s'agit de nettoyer et préparer les données pour la sélection de caractéristiques. La deuxième étape est la construction de vocabulaire des mots uniques. La troisième étape est la sélection des caractéristiques se fait en deux phase : La première est la sélection par le gain d'information : pour chaque caractéristique, nous calculons le gain d'information, puis nous sélectionnons le meilleur sous ensemble conduisant au meilleur score de classification. La deuxième phase consiste à utiliser le sous-ensemble de features obtenu dans la première phase pour sélectionner les features non redondants en utilisant un algorithme génétique.

4.3.1 Prétraitement

Le prétraitement est une étape importante avant la sélection de caractéristiques pour améliorer la qualité des données [54] afin de maximiser les chances de sélectionner les caractéristiques les plus informatives pour la tâche de classification.

a) **Elimination des mots vides :** les mots vides sont des mots très courants dans toutes les langues. En les supprimant, nous éliminons les informations de bas niveau et nous donnons plus de poids aux informations importantes dans le texte. Cette suppression réduit la taille de l'ensemble de données et par conséquent le temps d'entraînement du modèle, car il y a moins de tokens à traiter [55]. Certaines de ces mots sont :

- Les conjonctions de coordination (for, and, nor, but, or, yet , so, etc.).
- Les déterminants (a/an, the, this, that, these, those, etc.).
- Les prépositions (at, in, to, etc.).

Nous avons utilisé une méthode simple pour détecter et éliminer les mots vides en utilisant une liste de mots vides en anglais. Si l'un des tokens obtenus lors de la phase de tokénisation est présent dans cette liste, il sera supprimé du texte [56].

b) **La conversion en minuscules** : La conversion des termes du texte en lettre minuscules. Par exemple, "Bonjour" et "bonjour" devraient être considérés comme des mots identiques pour éviter d'introduire de redondance dans les données textuelles.

c) **Suppression des caractères spéciaux** : C'est l'opération de supprimer les éléments qui ne contribuent pas au sens du texte, tels que les nombres, la ponctuation, et les espaces supplémentaires tels que l'espace blanc. De plus, les balises HTML sont également supprimées car elles peuvent affecter la lisibilité du texte.

d) **La tokénisation :** C'est le processus de décomposition d'un texte en unités plus petites appelées tokens (jetons). Dans notre travail, nous avons utilisé la tokenisation Uni-gramme, qui est un type spécifique de tokenisation de mots qui divise un texte en mots ou tokens individuels, sans les regrouper en unités plus larges. Chaque mot dans le texte est considéré comme un token ou uni-gramme distinct.

e) **La lemmatisation :** C'est le processus qui consiste à ramener les mots d'un texte à leur forme canonique ou lemme. Après avoir séparé le texte en mots individuels, chaque mot est transformé en sa forme canonique en utilisant un ensemble de règles de morphologie et de dictionnaire. De cette façon, les formes fléchies d'un mot telles que les verbes conjugués ou les noms pluriels sont transformées en leur forme canonique.

4.3.2 La construction du vocabulaire des mots uniques

Après avoir appliqué les étapes de prétraitement au corpus de documents, la prochaine étape consiste à construire un vocabulaire de mots uniques.

Pour construire le vocabulaire, nous commençons par initialiser un ensemble vide. Ensuite, nous parcourons chaque document prétraité du corpus, et pour chaque document, nous parcourons la liste des mots. Pour chaque mot, nous vérifions s'il est déjà présent dans l'ensemble de vocabulaire. S'il n'y est pas, nous l'ajoutons à l'ensemble. À la fin de ce processus, l'ensemble contiendra tous les mots uniques présents dans le corpus prétraité.

4.3.3 Sélection basée sur le gain d'information

Le gain d'information (ou Information Gain – IG en anglais) est une méthode largement utilisée pour la sélection des caractéristiques pertinentes en apprentissage automatique. Le processus implique le calcul de l'IG pour chaque caractéristique dans l'ensemble de données et les classer dans l'ordre décroissant d'importance. Les caractéristiques avec un IG plus élevé

sont considérées comme plus pertinentes pour la tâche de classification et peuvent être sélectionnées à la fin de cette étape.

L'un des principaux avantages de l'utilisation de l'IG pour la sélection de caractéristiques est qu'elle aide à réduire la dimensionnalité des données et à se concentrer sur les caractéristiques les plus informatives.

Cependant, il y a quelques critiques de la méthode IG:

- IG considère le pouvoir prédictif individuel d'une caractéristique et ne prend pas en compte sa relation avec d'autres caractéristiques. Les caractéristiques ayant les scores IG les plus élevés sont représentatives, mais pas nécessairement de toutes les catégories. Lorsque le nombre de caractéristiques est petit, les caractéristiques sélectionnées peuvent ne pas couvrir toutes les catégories.

- La sensibilité du gain d'information (IG) au nombre de catégories dans un ensemble de données. Lorsqu'un ensemble de données ne contient que quelques catégories (deux ou trois), IG est susceptible d'identifier des caractéristiques qui ne sont pas biaisées en faveur d'une catégorie particulière. Cependant, à mesure que le nombre de catégories dans l'ensemble de données augmente, IG peut négliger des caractéristiques discriminantes importantes pour une ou deux catégories.

- La déviation de catégories mesure la répartition des catégories. Les catégories peuvent être classées en deux classes : la catégorie majoritaire (contient le plus grand nombre de documents) et la catégorie minoritaire (qui en contient le moins). Le gain d'information souffre de ce problème, qui se produit lorsque les données attribuées à différentes catégories dévient considérablement important, entraînant la suppression du pouvoir prédictif des caractéristiques représentatives pour les catégories mineures par une grande quantité de caractéristiques dans les catégories majeures. Ce problème affecte considérablement les méthodes de sélection de caractéristiques qui ne tiennent pas compte des relations entre termes, car les caractéristiques discriminantes peuvent être classées loin de leur position principale [57].

Malgré ces critiques, l'IG reste une méthode populaire pour la sélection de caractéristiques en raison de sa simplicité et de sa facilité de mise en œuvre. Elle est particulièrement utile pour des tâches telles que la classification de texte.

Rappelons de la formule du IG :

$$IG(t) = -\sum_{i=1}^{|c|} P(c_i) logP(c_i) + P(t) \sum_{i=1}^{|c|} P(c_i|t) logP(c_i|t) + P(\overline{t}) \sum_{i=1}^{|c|} P(c_i|\bar{t}) logP(c_i|\bar{t}) \quad (4.1)$$

Où $c_i$représente la ième catégorie, P(c_i) est la probabilité de la ième catégorie, P(t) et P($\bar{t}$) sont les probabilités que le terme t apparaît ou non dans l'ensemble des documents, respectivement, P(c_i| t) est la probabilité conditionnelle de la ième catégorie étant donné que le terme t est apparu, et P($c_i|\bar{t}$) est la probabilité conditionnelle de la ième catégorie étant donné que le terme t n'est pas apparu.

Pour calculer le gain d'information pour un terme, nous avons besoin de calculer l'entropie conditionnelle de la variable classe pour deux sous-ensembles de documents : l'un où le terme t est présent et l'autre où le terme t est absent. Pour ce faire, nous avons besoin des probabilités des classes pour chaque sous-ensemble de documents.

L'algorithme suivant résume les étapes de calcul de IG :

Algorithme Scores_IG

Entrées :C : domaine de l'étiquette de classe, E : domaine des valeurs d'un attribut.

Sorties : IG_valeurs (dictionnaire).

1. Intialization ;
2. S ,Som1, Som2, M, N = 0;
3. Pour chaque c_i de C:
4. Calculer P(c_i) ;
5. Hc = -(S+P(c_i) * log(P(c_i)));
6. S ← Hc ;
7. IG_valeurs = {} ;
8. pour $e_j$de E:
9. calculer P(e_j) , calculer P non(e_j) ;
10. fin pour
11. pour $e_j$de E
12. pour $c_i$de C
13. calculer P($c_i|e_j$) , calculer P(c_i|non e_j)
14. M=-(s1+P($c_i|e_j$)*log P($c_i|e_j$))
15. N=-(s2+P(c_i|non e_j)*log P (c_i|non e_j))..

16. Som1← M ;
17. Som2← N ;
18. Fin pour
19. IG = Hc - P (e_j) * M – P non (e_j) * N ;
20. IG_valeurs[e_j] = IG ;
21. Fin pour
22. Renvoyer IG_valeurs ;

Après le calcul des scores IG pour toutes les caractéristiques et leur organisation du plus élevé au plus petit, un sous-ensemble de termes est sélectionné en fonction d'un certain pourcentage des scores IG les plus élevés. Ce sous-ensemble est ensuite utilisé pour entraîner le classificateur, et son accuracy est mesurée sur un ensemble de test retenu. Ce processus est répété pour différents pourcentages des scores IG les plus élevés, et le sous-ensemble de termes qui donne le score du accuracy le plus élevé est sélectionné comme meilleur sous-ensemble.

4.3.4 Sélection basée sur les algorithmes génétiques

L'objectif de cette étude est d'améliorer la performance de la classification en sélectionnant les caractéristiques les plus importantes pour la classification des textes. Cependant, la méthode traditionnelle utilisée pour identifier les termes importants (IG) ne prend pas en compte la grande dimensionnalité de l'espace des caractéristiques. Cela rend impossible l'évaluation de tous les sous-ensembles possibles de l'ensemble de caractéristiques obtenu par la méthode IG. Une méthode de sélection de caractéristiques basée sur l'algorithme génétique (GA) est utilisée pour trouver le sous-ensemble optimal de caractéristiques qui maximise la performance de classification.

Avant de mettre en place l'algorithme génétique, il est essentiel de s'assurer que le problème à résoudre est adapté à cette méthode d'optimisation. Cela implique de définir les éléments de base de l'algorithme tels que les gènes, les chromosomes et la population, et de choisir judicieusement le codage, la sélection, le croisement, la mutation et les fonctions de fitness pour garantir une optimisation efficace.

Le tableau suivant présente les paramètres de l'algorithme génétique utilisés dans notre travail.

Paramètres	Valeurs
Taille de la population	80
Technique de sélection	truncation
Type de croisement	Un seul point
Taux de mutation	0.005
Nombre d'itérations	200

Tableau 4.1. Paramètres de notre algorithme génétique.

Les détails de notre algorithme génétique sont les suivants :

Encodage des individus : nous avons choisis le type d'encodage binaire car ce type permet de représenter de manière concise les ensembles de caractéristiques candidats. Une chaîne binaire appelée chromosome est utilisée pour représenter un ensemble de caractéristiques candidat. Les chromosomes, qui constituent la population, sont codés sous forme de vecteur binaire, et leurs gènes correspondent au nombre de caractéristiques présentes dans chaque espace de caractéristiques. En particulier, chaque bit du chromosome représente la présence ou l'absence d'une caractéristique spécifique. Cela permet de réduire la taille de la représentation et facilite le traitement des individus.

Initialisation de la population : La population est initialisée en utilisant une méthode alternative. Dans cette approche, nous attribuons une valeur de 1 ou 0 à chaque caractéristique du chromosome en fonction d'un modèle alterné qui commence par 1 ou 0. Les chromosomes sont composés de gènes, représentant chaque caractéristique du problème, codés en **1** si la caractéristique est sélectionnée et en **0** si elle ne l'est pas. Dans l'algorithme proposé, la longueur de chaque chromosome est égale au nombre de caractéristiques sélectionnées par IG.

Exemple illustratif :

Chromosome 1	**1**	**0**	**1**	**0**	**1**	**0**
Chromosome 2	**0**	**1**	**0**	**1**	**0**	**1**

Figure 4.1-Exemple illustratif sur l'initialisation.

En utilisant ce modèle alterné de valeurs 1 et 0, chaque chromosome de la population est différent des autres. De plus, l'utilisation de l'encodage binaire pour les chromosomes et les

gènes simplifie la représentation du problème et permet une manipulation et un croisement faciles pendant l'algorithme génétique.

La fonction de fitness : La fonction de fitness est utilisée pour évaluer la pertinence des individus dans la recherche de la solution optimale d'un problème. Chaque individu a une valeur de fitness propre. Une valeur de fitness élevée signifie que l'individu convient mieux comme solution au problème, tandis qu'une valeur de fitness plus faible signifie que l'individu convient moins bien comme solution au problème.

Cette étape consiste à optimiser la fonction de fitness en recherchant les chromosomes encodés, une fois que la population a été initialisée. Nous évaluons la fitness de chaque chromosome en sélectionnant les caractéristiques spécifiées par le chromosome, en entraînant un classificateur (naïve bayes par exemple) sur l'ensemble de données en utilisant ces caractéristiques et en calculant Accuracy (formule 4.3) du classificateur sur un ensemble de test. Le score de fitness de chaque chromosome est ensuite représenté par son score d'accuracy. Ensuite les valeurs de retour de la fonction de fitness sont triées du plus petit au plus grand selon accuracy.

$$Accuracy = \frac{VP + VN}{VP + FP + VN + FN} \qquad (4.3)$$

Où :

VP (Vrais positifs): le nombre d'observations positives correctement identifiées.

FP (Faux positifs): le nombre d'observations négatives identifiées à tort comme positives.

VN (Vrais négatifs) : le nombre d'observations négatives correctement identifiées.

FN (Faux négatifs): le nombre d'observations positives identifiées à tort comme négatives.

Nous avons choisi Accuracy comme une fonction de fitness, car elle est facile à interpréter et se concentre sur les performances globales.

Sélection : C'est un processus de reproduction qui a pour objectif de choisir les individus les plus performants de la population actuelle pour être transmis à la génération suivante. Dans la méthode de sélection truncation que nous avons opté, les 10 individus ayant les meilleurs scores de fitness calculés précédemment sont choisis pour constituer la nouvelle population.

Exemple illustratif : Supposons que nous avons une population de 15 individus, et que chaque individu a un score de fitness qui est calculé auparavant, et trié par ordre décroissant en fonction de leurs scores :

Individu A	95
Individu B	90
Individu C	85
Individu D	80
Individu E	75
Individu F	70
Individu G	65
Individu H	60
Individu I	55
Individu J	50
Individu K	45
Individu L	40
Individu M	30
Individu N	35
Individu E	20

Population actuelle

Truncation →

Individu A	95
Individu B	90
Individu C	85
Individu D	80
Individu E	75
Individu F	70
Individu G	65
Individu H	60
Individu I	55
Individu J	50

Nouvelle population

Figure 4.2- Sélection des individus pour la reproduction.

Pour sélectionner les 10 meilleurs individus, nous sélectionnerions les individus A à J, car ils ont les scores de fitness les plus élevés. Ces 10 individus seraient utilisés comme parents pour créer la nouvelle génération pour la prochaine itération.

Cette méthode est efficace en raison de sa simplicité et de son efficacité à sélectionner les individus les plus performants pour la reproduction.

Croisement : D'abord nous sélectionnons les individus à travers un processus d'accouplement. Ensuite pour former la nouvelle génération en adaptant la méthode de croisement, ce qui implique la création de deux nouveaux individus à partir des deux chromosomes existants. Dans notre travail, nous avons opté la technique à un seul point (one

point crossover). Le point de croisement est défini pour être au milieu du chromosome. Les valeurs des descendants sont changées après le point médian.

Exemple illustratif : Nous allons effectuer un croisement à un seul point au milieu des chromosomes, qui est le 4ème bit. Les offsprings résultants seront :

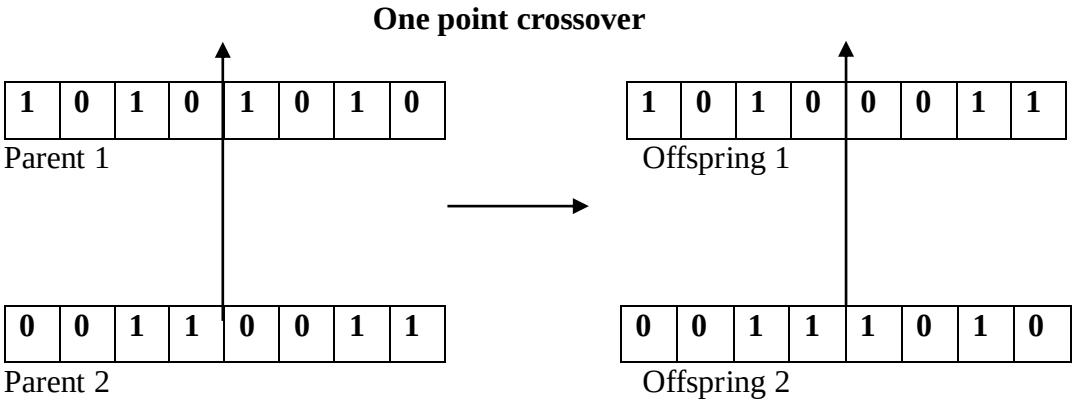

Figure 4.3- Un exemple illustratif sur le croisement.

Le 1er offspring est créé en prenant la première moitié du chromosome du parent 1 (1010) et la deuxième moitié du chromosome du parent 2 (0011) et en les concaténant. Le 2ème offspring est créé en prenant la première moitié du chromosome du parent 2 (0011) et la deuxième moitié du chromosome du parent 1 (1010) et en les concaténant.

Le choix d'un taux de croisement 0.8 élevé augmente la diversité génétique en générant des offsprings avec des combinaisons variées des caractéristiques parentales, ce qui prévient la convergence prématurée vers des solutions sous-optimales et favorise l'exploration de l'espace des solutions.

La mutation : La mutation est un processus qui permet d'introduire des variations locales chez les individus pour diversifier les chromosomes utilisés dans le processus de croisement. Cela aide à trouver des solutions différentes.

Pour effectuer l'opération de mutation, d'abord nous devons calculer le rang de mutation (mutation range) qui représente le nombre maximum de gènes pouvant être mutés chez un individu. Pour faire cela nous utilisons le taux de mutation qui présente la proportion de chromosomes qui subiront une mutation et le nombre de caractéristiques ou de gènes dans chaque chromosome. En multipliant ces deux valeurs, la valeur de rang de mutation **n** est

obtenue**.** Cela signifie que pour chaque chromosome, nous sélectionnerons aléatoirement n gènes à muter, en inversant leur valeur de 0 à 1 ou de 1 à 0.

Exemple illustratif : Supposons que nous avons une population de 5 individus, chacun avec un chromosome de longueur 10 représentant 10 caractéristiques différentes. Supposons que le taux de mutation est de 0,1, ce qui signifie qu'il y a 10% de chances qu'un gène du chromosome d'un individu soit muté.

Calcul du rang de mutation : 0.1 * 10 = 1. Donc le nombre maximum de gènes pouvant être mutés chez un individu est de 1.

Pour appliquer une mutation à cet individu, nous sélectionnerions au hasard un gène du chromosome à muter. Supposons que nous sélectionnions le 4ème gène, qui a une valeur de 0. Comme le gène doit être muté, nous inverserions sa valeur en 1.

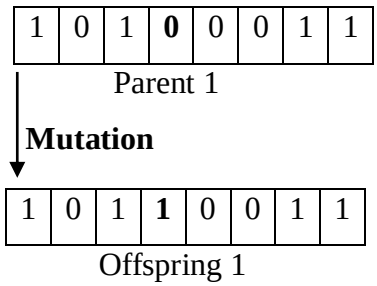

Figure 4.4. Un exemple illustratif sur la mutation.

Les choix de taux de mutation faible et de rang de mutation limité visent à trouver un équilibre entre exploration et exploitation, cela ainsi permet de préserver les solutions déjà performantes pour introduire de diversité et les caractéristiques prometteuses déjà présentes dans la population tout en permettant des variations locales pour explorer de nouvelles combinaisons

Les enfants générés par croisement et mutation sont ajoutés à la prochaine génération pour créer la population de la prochaine itération de l'algorithme génétique.

Le nombre de générations est fixé comme critère d'arrêt dans notre étude. L'algorithme continue d'itérer ce processus pour un nombre spécifié de générations, en mettant à jour à chaque fois la population avec un nouvel ensemble de chromosomes. L'itération de

l'algorithme s'arrête lorsque le critère d'arrêt est atteint. Après l'arrêt de l'algorithme, nous obtenons les meilleures caractéristiques représentées par les chromosomes qui ont les scores de fitness les plus élevés.

Le pseudo-code suivant résume le déroulement de l'algorithme génétique :

Algorithme algorithme-génétique

1. **Intialization**
2. Génération=0 ; n_parent = 10 ; max_géné = 200 ;génération_suivante ← {}
3. Initialiser population ;
4. While génération < max_géné
5. Evaluer le score de fitness du population
6. **Pour** i de 1 à 10
7. Sélectionner parents → parent;
8. Insérer parent dans génération_suivante ;
9. Fin pour
10. **Pour** i de 0 à taille (génération_suivante) - 1 pas == 2
11. Croissement → fils ;
12. Insérer fils dans génération_suivante ;
13. Fin pour
14. **Pour** i de 0 à taille(génération_suivante) pas == 1
15. Mutation → fils ;
16. Insérer fils dans génération_suivante ;
17. Fin pour
18. Mettre à jour la population actuelle ;
19. Génération++ ;
20. Fin while ;
21. Renvoyer Genération_suivante ;

4.5. Conclusion

Dans ce chapitre, nous avons présenté une méthode de sélection de caractéristiques pour la classification des textes, basée sur les algorithmes génétiques et l'information mutuelle (information gain). La méthode proposée vise à enlever les caractéristiques redondantes sélectionnées par IG en vue d'augmenter la performance du classifier tout en minimisant le

temps d’exécution. Nous avons donné un aperçu complet sur les étapes de prétraitement nécessaires pour les tâches de classification de texte, telles que le nettoyage, la tokenization et la lemmatization. Nous avons aussi montré comment construire un vocabulaire de mots uniques et discuté l'importance et les inconvénients du gain d'information (IG) et avons démontré comment l'utiliser pour sélectionner un sous-ensemble des termes les plus informatifs. Nous avons également montré comment nous pouvons optimiser davantage le processus de sélection de features en appliquant des algorithmes génétiques pour affiner l'ensemble de caractéristiques nécessaire à la tâche de classification.

Chapitre 5

Expérimentation et évaluation.

5.1 Introduction

Dans ce chapitre, nous présentons la configuration expérimentale ainsi que les résultats d'évaluation de notre méthode hybride. Nous décrivons l'environnement et les bibliothèques utilisés pour la mise en œuvre, discutons les étapes de prétraitement appliquées aux données textuelles et présentons les détails de l'implémentation de notre méthode. De plus, nous comparons les performances de la méthode hybride avec les méthodes individuelles (IG, MI, CH2, et IGI) afin d'évaluer son efficacité dans la capture de caractéristiques (features) pertinentes.

5.2 Description d'enivrements et de bibliothèques

5.2.1 L'enivrement utilisé

Python : C'est un langage de programmation puissant, très apprécié pour sa sémantique dynamique et sa nature orientée objet. Il excelle dans le développement rapide d'applications et sert de langage de script efficace pour connecter divers composants. La syntaxe de Python est conçue pour être conviviale, mettant l'accent sur la lisibilité et réduisant la charge de maintenance des programmes. Il favorise la modularité et la réutilisation du code grâce à sa prise en charge des modules et des packages. Python est largement accessible, avec l'interpréteur Python et une vaste bibliothèque standard disponibles gratuitement sur les principales plateformes. L'un de ses principaux avantages est la productivité accrue qu'il offre, grâce à l'absence d'étape de compilation. De plus, le débogage des programmes Python est simplifié car il génère des exceptions au lieu de provoquer des erreurs de segmentation [62].

Jupyter Notebook : C'est une application web open source utilisée pour la création et le partage de documents contenant du code interactif, des équations, des visualisations et du texte. Il est maintenu par l'équipe de Project Jupyter [63].

5.2.2 les bibliothèques nécessaires

Scikit-learn (Sklearn) : C'est la bibliothèque la plus utile et solide pour l'apprentissage automatique en Python. Elle offre une sélection d'outils efficaces pour l'apprentissage automatique et la modélisation statistique, y compris la classification, la régression, le regroupement et la réduction de la dimensionnalité via une interface cohérente en Python. Cette bibliothèque, largement écrite en Python, est basée sur NumPy, SciPy et Matplotlib [64].

Matplotlib : C'est une bibliothèque Python utilisée pour créer des graphiques et des tracés en 2D à l'aide de scripts Python. Elle dispose d'un module appelé pyplot qui facilite la création de graphiques en offrant des fonctionnalités pour contrôler les styles de lignes, les propriétés de police, le formatage des axes, etc. Matplotlib prend en charge une grande variété de graphiques et de tracés tels que les histogrammes, les graphiques à barres, les spectres de puissance, les graphiques d'erreur, etc. Elle est utilisée en association avec NumPy pour fournir un environnement qui constitue une alternative open source efficace à MatLab. Elle peut également être utilisée avec des kits d'outils graphiques tels que PyQt et wxPython [65].

Pandas : C'est une bibliothèque Python open-source utilisée pour la manipulation de données à haute performance et l'analyse de données à l'aide de ses puissantes structures de données. Python avec Pandas est utilisé dans divers domaines académiques et commerciaux, tels que la finance, l'économie, les statistiques, la publicité, l'analyse web, et bien d'autres. Grâce à Pandas, nous pouvons accomplir cinq étapes typiques dans le traitement et l'analyse des données, quel que soit leur origine : chargement, organisation, manipulation, modélisation et analyse des données [66].

NLTK : C'est une plateforme utilisée pour développer des programmes Python qui traitent les données linguistiques humaines dans le cadre du traitement automatique du langage naturel (TALN). Il contient des bibliothèques de traitement de texte pour la tokenisation, l'analyse syntaxique, la classification, la racinisation, l'étiquetage et le raisonnement sémantique [67].

NumPy : est une bibliothèque Python open source utilisée dans presque tous les domaines de la science et de l'ingénierie. Elle constitue la norme universelle pour travailler avec des données numériques en Python et elle est au cœur de l'écosystème scientifique de Python [68].

5.3 Présentation et prétraitement des jeux de données

5.3.1 Description de jeux de données

Restaurant Reviews : C'est un jeu de donné utilisé pour le traitement du langage naturel. Ce jeu de données contient deux colonnes : "Avis des clients" et "Aimé". Les avis des clients nous informent sur les critiques données par les clients pour une nourriture dans un restaurant, et la colonne "Aimé" indique s'ils ont aimé la nourriture ou non [69]. La figure ci-dessus présente un aperçu général pour Restaurant Reviews :

	text	category
0	Wow... Loved this place.	1
1	Crust is not good.	0
2	Not tasty and the texture was just nasty.	0
3	Stopped by during the late May bank holiday of...	1
4	The selection on the menu was great and so wer...	1
...	...	...
995	I think food should have flavor and texture an...	0
996	Appetite instantly gone.	0
997	Overall I was not impressed and would not go b...	0
998	The whole experience was underwhelming, and I ...	0
999	Then, as if I hadn't wasted enough of my life ...	0

1000 rows × 2 columns

Figure 5.1-Un aperçu du dataset Restaurant.

Fake News : Ce jeu de données contient une liste d'articles considérés comme des "fausses" informations. Collecté pour la classification des informations (news) [6]. La figure ci-dessus présente un aperçu général pour Fake News:

	text	category
0	Donald Trump just couldn t wish all Americans ...	News
1	House Intelligence Committee Chairman Devin Nu...	News
2	On Friday, it was revealed that former Milwauk...	News
3	On Christmas day, Donald Trump announced that ...	News
4	Pope Francis used his annual Christmas Day mes...	News
...	...	...
23476	21st Century Wire says As 21WIRE reported earl...	Middle-east
23477	21st Century Wire says It s a familiar theme. ...	Middle-east
23478	Patrick Henningsen 21st Century WireRemember ...	Middle-east
23479	21st Century Wire says Al Jazeera America will...	Middle-east
23480	21st Century Wire says As 21WIRE predicted in ...	Middle-east

23481 rows × 2 columns

Figure 5.2- Un aperçu du dataset Fake News.

5.3.2 Prétraitement

Comme nous l'avons mentionné dans le chapitre 4, le prétraitement est une étape essentielle pour les tâches de classification, et les différentes étapes que nous avons appliquées sont les suivantes :

1. Convertir tous les textes en minuscules.
2. Supprimer les balises HTML.
3. Supprimer les chiffres.
4. Supprimer les caractères spéciaux.
5. Supprimer les mots de moins de 4 caractères.
6. Supprimer les espaces supplémentaires entre les mots.
7. Supprimer les espaces à gauche et à droite.
8. Supprimer la ponctuation.
9. Tokeniser les mots.
10. Lemmatisation.
11. Supprimer les mots vides (stop words).

Les figure ci-dessus présentent les deux datasets après prétraitement :

	text	category
0	loved place	1
1	crust good	0
2	tasty texture nasty	0
3	stopped late bank holiday rick steve recommend...	1
4	selection menu great price	1
...	...	...
995	think food flavor texture lacking	0
996	appetite instantly gone	0
997	overall impressed would back	0
998	whole experience underwhelming think ninja sus...	0
999	wasted enough life poured salt wound drawing t...	0

1000 rows × 2 columns

Figure 5.3- Un aperçu du dataset Restaurant après prétraitement.

	text	category
0	donald trump wish american happy year leave in...	News
1	house intelligence committee chairman devin nu...	News
2	friday revealed former milwaukee sheriff david...	News
3	christmas donald trump announced would back wo...	News
4	pope francis used annual christmas message reb...	News
...	...	...
23476	century wire say wire reported earlier week un...	Middle-east
23477	century wire say familiar theme whenever dispu...	Middle-east
23478	patrick henningsen century wireremember obama ...	Middle-east
23479	century wire say jazeera america history bigge...	Middle-east
23480	century wire say wire predicted year look ahea...	Middle-east

23481 rows × 2 columns

Figure 5.4- Un aperçu du dataset Fake News après prétraitement.

5.4 Classification

5.4.1 Classification sans sélection de features

Dans cette section, notre objectif est d'appliquer la classification sans sélection de caractéristiques et de la comparer avec la classification utilisant la sélection de caractéristiques. En comparant les résultats des deux classifications, nous visons à démontrer l'efficacité de la technique de sélection de caractéristiques.

Le tableau ci-dessus présente le résultat de classification en fonction d'accuracy sans sélection de features en utilisant les deux classificateurs NB et SVM :

Classifeur	NB	SVM
	Accuracy	Accuracy
Fake News	0.572	0.568
Restaurant	0.764	0.756

Tableau .5.1. Résultats de classification sans sélection de features.

5.4.2 Classification avec sélection des features par MI, CH2, IGI et IG

Les tableaux ci-dessous présentent les performances de classification de différents classificateurs, notamment Naive Bayes (NB) et Support Vector Machine (SVM), en utilisant des méthodes de sélection de caractéristiques telles que l'information mutuelle (MI), le test chi2 (CH2), l'indice de gini amélioré (IGI) et le gain d'infomation (IG) :

Classifeurs	Métriques							
	MI		CH2		IGI		IG	
	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy
NB	710	0.568	483	0.788	28	0.468	454	0.764
SVM	483	0.572	483	0.792	653	0.644	398	0.788

Tableau 5.2. Résultats classification avec sélection de features pour le dataset Fake News.

Classifeurs	Métriques							
	MI		CH2		IGI		IG	
	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy
NB	404	0.684	404	0.824	404	0.512	323	0.832
SVM	404	0.624	177	0.832	404	0.512	161	0.836

Tableau 5.3. Résultats classification avec sélection de features pour le dataset Restaurent.

- Dans le tableau de comparaison fourni pour **Fake News**, les classificateurs (NB) et (SVM) ont obtenu leurs meilleures accuracy avec la métrique IG. NB a atteint un score de 0,764 avec 454 termes, tandis que SVM a obtenu un score de 0,788 avec seulement 398 termes.

- Dans le deuxième tableau de comparaison pour **Restaurant**, les résultats confirment la tendance observée dans le premier tableau, où les deux classificateurs ont également obtenu leurs meilleures accuracy avec la métrique IG. NB a atteint une accuracy de 0,832 avec 323 termes, tandis que SVM a obtenu une accuracy légèrement plus élevée de 0,836 avec seulement 161 termes.

5.4.3 Sélection avec l'algorithme génétique

Dans cette section nous allons utiliser les termes sélectionnés précédemment par les métriques MI, CH2, IGI et IG pour effectuer une sélection plus fine avec l'algorithme génétique, puis nous effectuons une classification par les deux classificateurs NB et SVM, ensuite nous présentons les différents résultats.

Les tableaux ci-dessous présentent les résultats de la meilleure accuracy obtenue pour chaque génération sur une période de 200 itérations, en utilisant SVM et NB comme classifieur :

Génération	**Métriques**							
	MI		**CH2**		**IGI**		**IG**	
	NB	**SVM**	**NB**	**SVM**	**NB**	**SVM**	**NB**	**SVM**
	accuracy							
1	0.52	0.536	0.684	0.712	0.332	0.588	0.7	0.688
2	0.524	0.548	0.688	0.72	0.34	0.592	0.704	0.704
3	0.532	0.548	0.692	0.716	0.34	0.592	0.708	0.704
4	0.532	0.552	0.696	0.728	0.34	0.592	0.712	0.708
5	0.536	0.552	0.7	0.728	0.34	0.592	0.708	0.708
-------	----------	--------	-------	----------	--------	----------	--------	---------
196	0.648	0.664	0.836	0.856	0.46	0.632	0.824	0.844
197	0.648	0.664	0.836	0.86	0.46	0.632	0.82	0.84
198	0.648	0.664	0.836	0.86	0.46	0.632	0.82	0.844
199	0.648	0.664	0.836	0.86	0.46	0.632	0.816	0.848
200	0.648	0.664	0.836	0.86	0.46	0.632	0.82	0.848

Tableau 5.4. Évolution de l'Accuracy par génération pour MI, CH2,IGI et IG avec GA pour Fake News

Génération	Métriques							
	MI		CH2		IGI		IG	
	NB	SVM	NB	SVM	NB	SVM	NB	SVM
	Accuracy							
1	0.62	0.604	0.68	0.74	0.516	0.516	0.74	0.76
2	0.628	0.608	0.704	0.756	0.516	0.516	0.748	0.772
3	0.628	0.612	0.708	0.756	0.516	0.516	0.744	0.772
4	0.632	0.612	0.724	0.756	0.516	0.516	0.744	0.772
5	0.636	0.616	0.724	0.756	0.516	0.516	0.76	0.772
-------	-------	------	--------	---------	---------	----------	---------	---------
196	0.792	0.716	0.892	0.88	0.516	0.516	0.88	0.88
197	0.792	0.716	0.884	0.884	0.516	0.516	0.884	0.876
198	0.792	0.716	0.884	0.884	0.516	0.516	0.884	0.88
199	0.792	0.716	0.884	0.884	0.516	0.516	0.884	0.876
200	0.792	0.716	0.884	0.888	0.516	0.516	0.884	0.88

Tableau 5.5. Évolution de l'Accuracy par génération pour MI, CH2, IGI et IG avec GA pour Restaurant.

D'après les résultats obtenus dans les tableaux 5.4 et 5.5, nous observons que les scores commencent généralement avec une valeur faible lors de la première génération, puis s'améliorent progressivement au fil des générations jusqu'à atteindre une valeur maximale et optimal. Ces résultats confirment l'efficacité de l'algorithme génétique pour améliorer les performances des classificateurs NB et SVM.

Nous fournissons un tableau complet qui présente l'accuracy de l'algorithme génétique et le nombre de termes pour chaque dataset, en utilisant toujours les classificateurs SVM et NB :

Classifeurs	Métriques							
	MI		CH2		IGI		IG	
	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy
NB	238	0.652	226	0.836	169	0.46	227	0.824
SVM	204	0.668	215	0.86	229	0.632	236	0.848

Tableau 5.6. Résultats classification pour (MI,CH2,IGI et IG) avec GA pour Fake News.

Classifeurs	Métriques							
	MI		CH2		IGI		IG	
	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy	Nb termes	accuracy
NB	223	0.792	220	0.896	215	0.516	251	0.884
SVM	228	0.72	226	0.888	213	0.516	225	0.884

Tableau 5.7. Résultats classification pour (MI, CH2, IGI et IG) avec GA pour Restaurant.

En comparant les résultats obtenus dans les tableaux 5.2 et 5.3 avec ceux des tableaux 5.6 et 5.7, nous observons que l'application de l'algorithme génétique sur les termes sélectionnés par les métriques (MI, CH2, IGI et IG) a conduit à un meilleur score que l'application de ces métriques individuellement, tout en minimisant le nombre de termes. Ces résultats confirment l'efficacité de l'algorithme génétique pour améliorer les performances des classificateurs NB et SVM, et son capacité à sélectionner de manière optimale les termes pertinents tout en réduisant la redondance**.**

5.5 Discussion

Les figures ci-dessous montrent les scores d'accuracy pour la tâche de classification en utilisant les classificateurs Naive Bayes et SVM avec les métriques IG, IGI, CH2 et MI pour les deux ensembles de données. Un seuil de 100 termes est ajouté à chaque itération, et le score d'accuracy est calculé et enregistré à chaque étape :

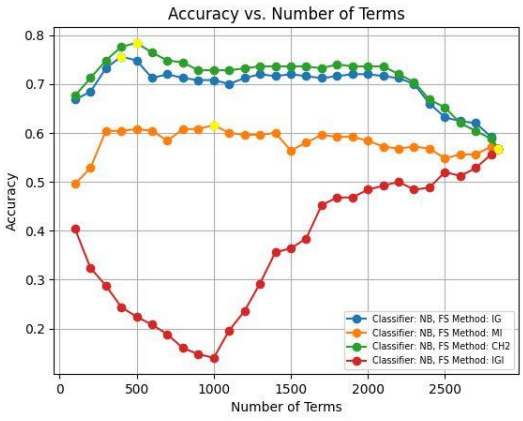

Figure 5.5. Résultats de classification de Fake avec NB.

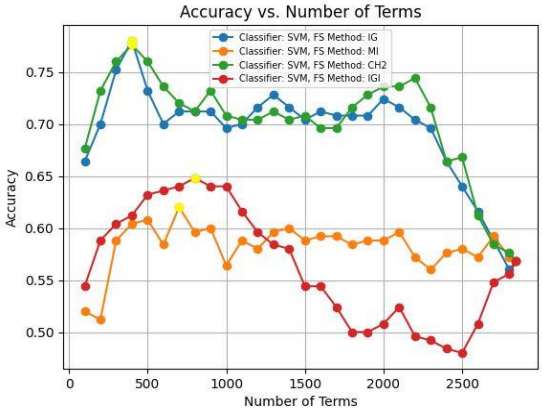

Figure 5.6. Résultats de classification de Fake avec SVM.

.

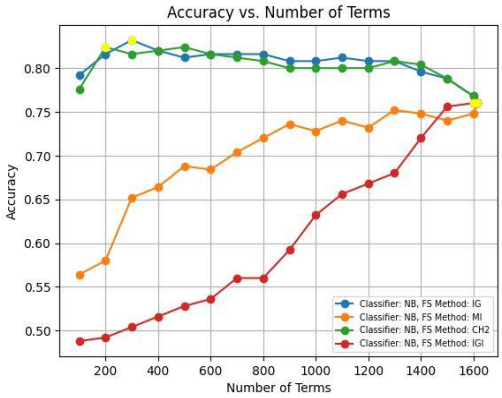

Figure 5.7. Résultats de classification de Restaurant avec NB

.

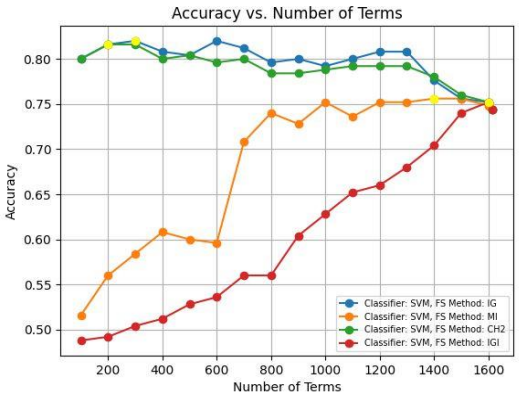

Figure 5.8. Résultats de classification de Restaurant avec SVM.

.

Les figures ci-dessous montrent les scores d'accuracy pour la tâche de classification en utilisant les classificateurs Naïve Bayes et SVM avec les métrique IG, MI, CH2, IGI et l'algorithme génétique pour les deux ensembles de données :

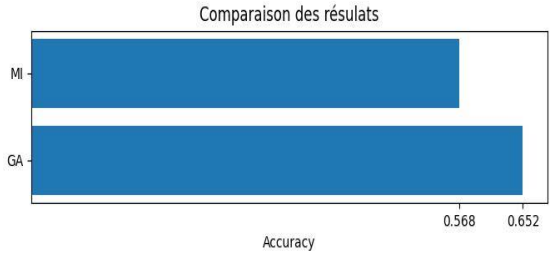

Figure 5.9. Résultats de classification de Fake News avec NB (MI-GA)

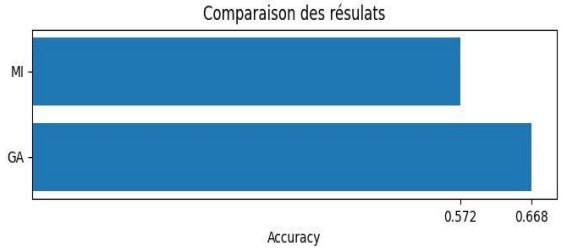

Figure 5.10. Résultats de classification de Fake News avec SVM (MI-GA)

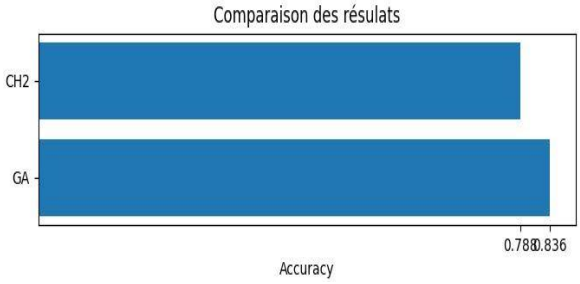

Figure 5.11. Résultats de classification de Fake News avec NB (CH2-GA)

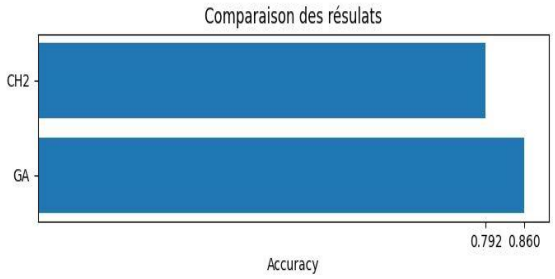

Figure 5.12. Résultats de classification de Fake News avec SVM (CH2-GA)

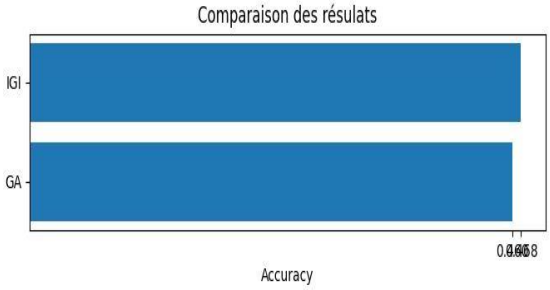

Figure 5.13. Résultats de classification de Fake News avec NB (IGI-GA)

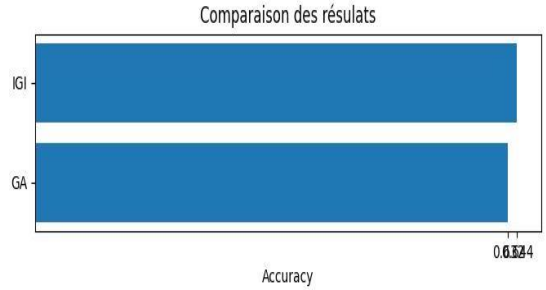

Figure 5.14. Résultats de classification de Fake News avec SVM (IGI-GA)

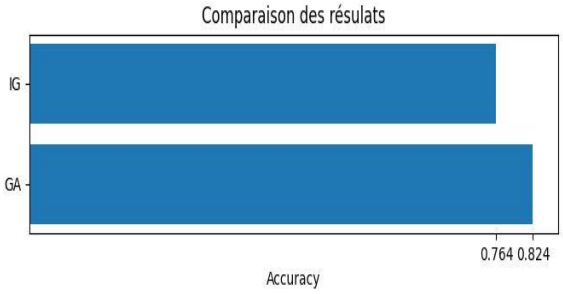

Figure 5.15. Résultats de classification de Fake News avec NB (IG-GA)

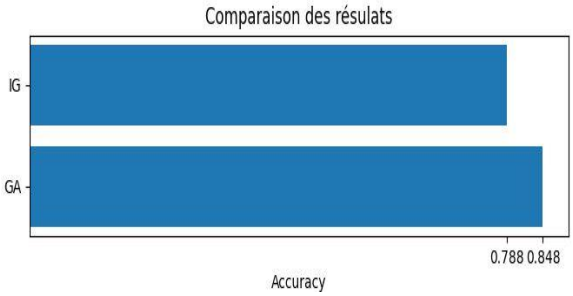

Figure 5.16. Résultats de classification de Fake News avec SVM (IG-GA)

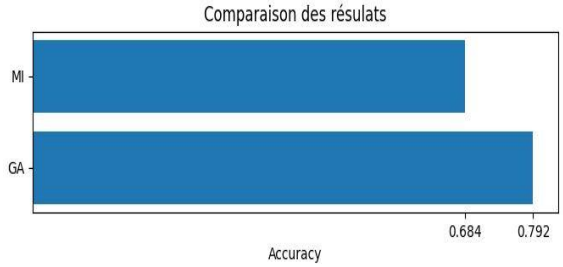

Figure 5.17. Résultats de classification de Restaurant avec NB (MI-GA)

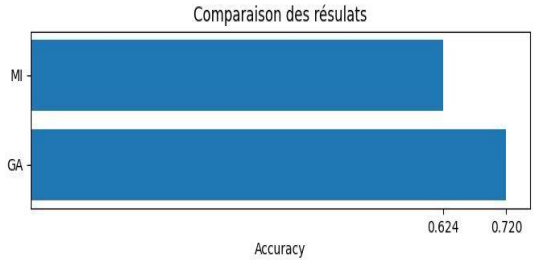

Figure 5.18. Résultats de classification de Restaurant avec SVM (MI-GA)

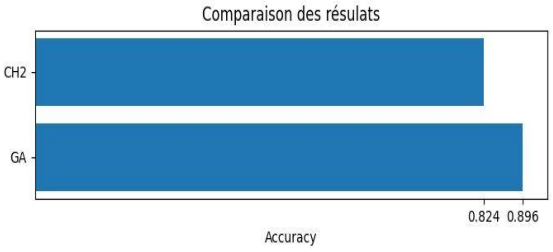

Figure 5.19. Résultats de classification de Restaurant avec NB (CH2-GA)

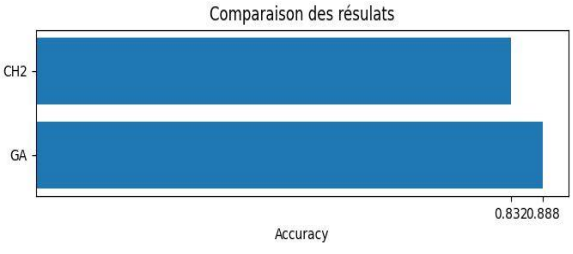

Figure 5.20. Résultats de classification de Restaurant avec SVM (CH2-GA)

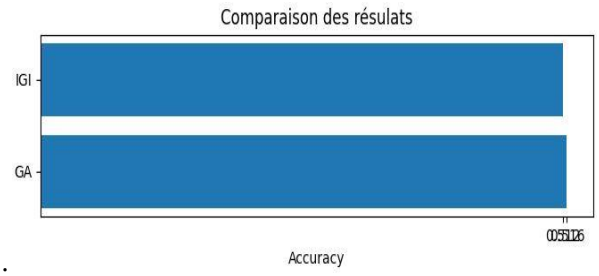

Figure 5.21. Résultats de classification de Restaurant avec NB (IGI-GA)

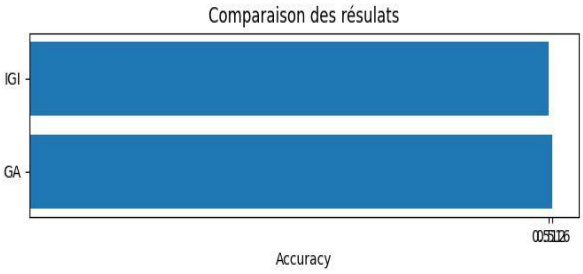

Figure 5.22. Résultats de classification de Restaurant avec SVM (IGI-GA)

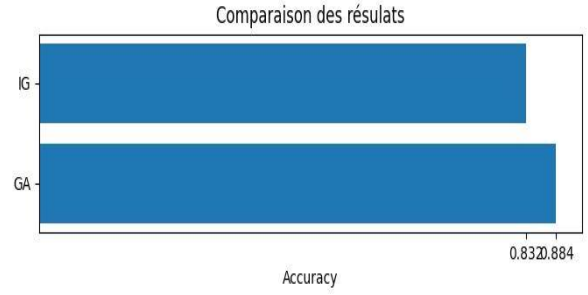

Figure 5.23. Résultats de classification de Restaurant avec NB (IG-GA)

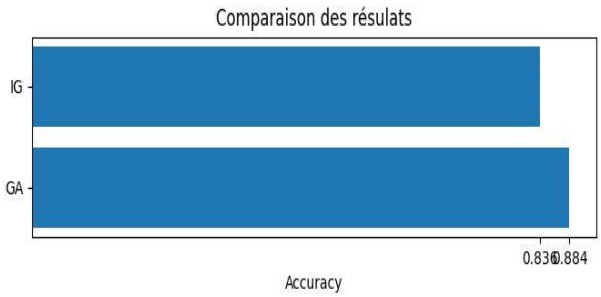

Figure 5.24. Résultats de classification de Restaurant avec SVM (IG-GA)

Les résultats obtenus à partir de notre méthode hybride, qui combine des méthodes de sélection de caractéristiques (IG, MI, CH2, et IGI) et notre algorithme génétique, ont fourni des informations précieuses sur les performances de la tâche de classification. Plus précisément, nous avons observé qu'IG surpassait les autres méthodes de sélection de caractéristiques, y compris l'information mutuelle (MI), CH2 et l'indice de Gini, en termes de réalisation d'une plus grande accuracy.

Cette constatation indique qu'IG était particulièrement efficace pour identifier et sélectionner les caractéristiques pertinentes pour la tâche de classification. La capacité d'IG à classer les caractéristiques en fonction de leur contenu informationnel et de leur pertinence par rapport à la variable cible a probablement contribué à sa performance supérieure. Ce résultat est cohérent avec des recherches antérieures qui ont mis en évidence l'efficacité d'IG en tant que méthode de sélection de caractéristiques.

De plus, l'inclusion d'un algorithme génétique dans notre approche hybride a encore amélioré les performances globales. L'algorithme génétique a exploité le pouvoir des principes évolutifs pour optimiser le processus de sélection de caractéristiques et affiner le sous-ensemble de caractéristiques. Par conséquent, l'algorithme génétique a pu identifier un sous-

ensemble de caractéristiques encore plus optimal que IG seul, ce qui a conduit à une amélioration de la précision de la classification.

5.6 Conclusion

En conclusion, l'implémentation de notre méthode hybride combinant les méthodes de sélection basées sur le filtre telles que IG, MI, CH2, etc., avec l'algorithme génétique pour la sélection de caractéristiques pours la classification de texte ont démontré sa supériorité par rapport aux méthodes individuelles. La méthode hybride offre une solution complète pour améliorer l'efficacité des méthodes basées sur le filtre, et se traduit par une amélioration des performances de sélection de caractéristiques en termes de performance de classification traduits par deux classificateurs NB et SVM.

Conclusion générale et perspectives

La classification de texte est un sujet bien connu et largement étudié dans le domaine de la recherche scientifique. Dans ce contexte, la sélection des caractéristiques est considérée comme une technique essentielle pour la tâche de classification des textes, où nous avons développé une approche hybride en combinant le gain d'information (IG) avec l'algorithme génétique.

Notre méthode hybride a été évaluée sur deux ensembles de données de textes et comparée à d'autres méthodes telles que le MI, CH2 et l'IGI. Les résultats de nos expériences ont démontré que notre méthode hybride était capable de sélectionner un sous-ensemble optimal de caractéristiques. Elle a significativement amélioré les performances de classification par rapport aux autres méthodes (MI, CH2 et IGI).

Cette approche hybride a permis de surmonter les limitations des méthodes traditionnelles en termes de redondance des caractéristiques sélectionnées. En combinant le calcul du gain d'information pour évaluer l'importance des caractéristiques avec l'algorithme génétique pour optimiser la sélection, notre méthode a pu obtenir de meilleurs résultats en termes d’accuracy et de temps de calcul.

Cette méthode hybride offre de nouvelles perspectives pour améliorer les performances des modèles de classification des textes. Des recherches supplémentaires pourraient être envisagées pour étendre cette méthode à d'autres ensembles de données et domaines d'application, afin de continuer à améliorer les résultats obtenus.

Bibliographie & Webographie

1. Biobligraphie

[1] Shen, D. (2009). Text Categorization. In: LIU, L., ÖZSU, M.T. (eds) Encyclopedia of Database Systems. Springer.

[3] Belainine Billal. Avril 2017. « Classification supervisée de textes courts et bruités: Application au domaine des médias sociaux », Mémoire de maîtrise ,Université du Québec à Montréal.

[4] Katakis, I. M., Tsoumakas, G., & Vlahavas, I. P. (2008). « Multilabel Text Classification for Automated Tag Suggestion », In Proceedings of the International Conference on *Data* Mining (pp. 917-922).

[6] Luque, A., Gómez-Bellido, J., Carrasco, A., & Barbancho, J. (2018). « Optimal Representation of Anuran Call Spectrum in Environmental Monitoring Systems Using Wireless Sensor Networks », Sensors, 18(6), 1803.

[7] Obi, Jude. (2023). « A Comparative Study of Several Classification Metrics and Their Performances on Data »,World Journal of Advanced Engineering Technology and Sciences, 8(1), 308-314.

[8] Alsmadi, Issa, & Hoon, Gan Keng. (2019). « Term weighting scheme for short-text classification: Twitter corpuses »,Neural Computing and Applications, 31(10), 3819-3831.

[9] George K, Soumya, & Joseph, Shibily. (Jan. 2014). « Text Classification by Augmenting Bag of Words (BOW) Representation with Co-occurrence Feature », IOSR Journal of Computer Engineering (IOSR-JCE), 16(1), 34-38.

[10] Kim, Sang-Woon, & Gil, Joon-Min. (2019). « Research paper classification systems based on TF-IDF and LDA schemes », Human-centric Computing and Information Sciences, 9(1), 30.

[11] Agarwal, N., Sikka, G., & Awasthi, L.K. (2020). « Enhancing Web Service Clustering using Length Feature Weight Method for Service Description Document Vector Space Representation », Expert Systems with Applications.

[12] Aggarwal, C.C., & Zhai, C. (2012). « A Survey of Text Classification Algorithms », Mining Text *Data*.

[13] Sheykhmousa, Mohammadreza & Mahdianpari, Masoud. (2020). « Support Vector Machine vs. Random Forest for Remote Sensing Image Classification: A Meta-analysis and systematic review », IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[16] Zhang, Wen & Yoshida, Taketoshi & Tang, Xijin. (2008). « Text classification based on multi-word with support vector machine », Knowledge-Based Systems, 21, 879-886.

[15] Gharib, Tarek & Badieh Habib Morgan, Mena & Fayed, Zaki. (2009). » Arabic Text Classification Using Support Vector Machines », I. J. Comput. Appl.. 16. 192-199.

[16] Yong, Zhou & Youwen, Li & Shixiong, Xia. (2009). « An Improved KNN Text Classification Algorithm Based on Clustering », Journal of Computers. 4.

[17] Ifeanyi-Reuben, N., Odikwa, N., & Ugwu, C. (2021). « N-gram and K-Nearest Neighbour Based Igbo Text Classification Model », International Journal of Innovative Science and Research Technology, 6, 759-766.

[18] Bhatia, Nitin & Vandana,. (2010). « Survey of Nearest Neighbor Techniques », International Journal of Computer Science and Information Security. 8.

[19] Priyanka and Kumar, D. (2020) . « Decision tree classifier: a detailed survey », Int. J. Information and Decision Sciences, Vol. 12, No. 3, pp.246–269.

[20] Rokach, L., & Maimon, O. (2005). Decision Trees. In: Maimon, O., Rokach, L.Data Mining and Knowledge Discovery Handbook, 165-192. Springer, Boston.

[21] Rochim, A. F., Kusumastuti, R., & Windasari, I. P. « Comparison of Feature Selection for Naïve Bayes Classification Method in A Case Study of The Coronavirus Lockdown ».

[22] Naseriparsa, Mehdi, Bidgoli, Amir Massoud, & Varaee, Touraj. (2014). « A Hybrid Feature Selection Method to Improve Performance of a Group of Classification Algorithms », International Journal of Computer Applications, 69.

[23] Suppers, Anouk, van Gool, Alain, & Wessels, Hans. (2018). « Integrated Chemometrics and Statistics to Drive Successful Proteomics Biomarker Discovery », Proteomes, 6(2), 20.

[24] Ouali, Choayb. « Classification automatique de textes », Mémoire de fin d'étude, Université de M'sila, 2014.

[25] Mladenić, D. (2011). « Feature Selection in Text Mining », In: Sammut, C., Webb, G.I. Encyclopedia of Machine Learning. Springer.

[26] Jovic, A., Brkić, K., & Bogunovic, N. (2015). « A review of feature selection methods with applications », In Proceedings of the 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (pp. 1200-1205).

[27] Khalid, S., Khalil, T., & Nasreen, S. (2014). « A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning », In Proceedings of the Science and Information Conference 2014 (pp. 372). London, UK.

[28] Shah, F.P., & Patel, V. (2016). « A Review on Feature Selection and Feature Extraction for Text Classification », In Proceedings of the IEEE WiSPNET 2016 Conference.

[29] Zhu, L., Wang, G., & Zou, X. (2017). Improved information gain feature selection method for Chinese text classification based on word embedding. In Proceedings of the 6th International Conference on Software and Computer Applications (pp. 72-76).

[30] Al-Harbi, O. (2019). « A Comparative Study of Feature Selection Methods for Dialectal Arabic Sentiment Classification Using Support Vector Machine », IJCSNS International Journal of Computer Science and Network Security, 19(1), pp. 167-176.

[54] Rachburee, N., & Punlumjeak, W. (2015). A Comparison of Feature Selection Approach Between Greedy, IG-ratio, Chi-square, and mRMR in Educational Mining. Presented at the 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE), Chiang Mai, Thailand.

[31] Deng, X., Li, Y., Weng, J. et al. (2019). « Feature selection for text classification: A review », Multimedia Tools and Applications, 78, 3797-3816.

[33] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R., Tang, J., & Liu, H. (2016). « Feature Selection: A Data Perspective », ACM Computing Surveys, 50.

[34] Khater, B.S.; Abdul Wahab, A.W.; Idris, M.Y.I.; Hussain, M.A.; Ibrahim, A.A.; Amin, M.A.; Shehadeh, H.A.(2021). « Classifier Performance Evaluation for Lightweight IDS Using Fog Computing in IoT Security », Electronics ,10 (14). p. 1633.

[35] Xie, L., Li, Z., Zhou, Y., He, Y., & Zhu, J. (2020). « Computational Diagnostic Techniques for Electrocardiogram Signal Analysis », Sensors.

[36] AZZOUG, Soraya. « Sélection Automatique des Caractéristiques pour la Reconnaissance des Chiffres Manuscrits par la Méthode F-score », Mémoire de magister, Université des Sciences et de la Technologie Houari Boumediene, 04 juillet 2013.

[37] S.N. Sivanandam, S.N. Deepa. « Introduction to Genetic Algorithms », Springer.

[41] Gomez, F., & Miikkulainen, R. (2002). « Robust Non-Linear Control through Neuroevolution ».

[38] Kumar, M., Husain, M., Upreti, N., & Gupta, D. (December 1, 2010). « Genetic Algorithm: Review and Application ».

[39] AbdulHamed, A. A., Tawfeek, M. A., Keshk, A. E. (2018). « A Genetic Algorithm for Service Flow Management with Budget Constraint in Heterogeneous Computing », Future Computing and Informatics Journal,3.

[45] Zerari, N. (2006). « Les algorithmes génétiques en maintenance », Mémoire de Magister, Université El Hadj Lakhdar Batna.

[46] Mehidid, F. 19/06/2013 « Algorithme Génétique .Mémoire de Master », Université Abdelhamid Ibn Badis-Mostaganem.

[44] Bourazza, S. (2006). « Variantes d'algorithmes génétiques appliquées aux problèmes d'ordonnancement » ,Thèse de Doctorat,Université du Havre.

[47] Shukla, A., Pandey, H. M., & Mehrotra, D. (2015). « Comparative review of selection techniques in genetic algorithm », In 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE).

[49] Yang, X.-S. (2014). « Nature-Inspired Optimization Algorithms ». Analysis of Algorithms (Chapter 2, p. 30).

[48] Yadav, Saneh & Sohal, Asha. (2017). « Study of the various selection techniques in Genetic Algorithms ».

[50] Ali Pitchay, Sakinah & Shorman, Samer. (2015). « Significance of parameters in genetic algorithm, the strengths, its limitations and challenges in image recovery », Journal of Engineering and Applied Science,10, 585-593.

[51] Joshi, Divya. (2021). « Genetic Algorithm and its Applications - A Brief Study ». ASIAN JOURNAL OF CONVERGENCE IN TECHNOLOGY. 7. 8-12.

[60] AbdulHamed, Ahmed & Tawfeek, Medhat & Keshk, Arabi. (2018). « A Genetic Algorithm for Service Flow Management with budget Constraint in Heterogeneous Computing. » Future Computing and Informatics Journal. 3.

[52] Osl, M., Dreiseitl, S., Cerqueira, F., Netzer, M., Pfeifer, B., Baumgartner, C. (2009). « Demoting redundant features to improve the discriminatory ability in cancer data », Journal of Biomedical Informatics, 42(4), 721-725.

[53] Bennasar, M., Hicks, Y., Setchi, R. (2015). « Feature selection using Joint Mutual Information Maximisation », Expert Systems with Applications, 42(22), 8520-8532.

[56] YOUSFI Yasmine, BELLAHOUES Yasmine. (2018/2019). « Sélection de Caractéristiques pour la Classification de Polarité d'Opinion », Mémoire De Fin D'étude De Master Professionnel,Université de Tizi-Ouzou.

[57] Wang, Gang & Lochovsky, Frederick. (2004). « Feature selection with conditional mutual information MaxiMin in text categorization », International Conference on Information and Knowledge Management, Proceedings. 342-349.

[58] Doddipalli, Lavanya & Rani, K.. (2011). « Analysis of feature selection with classification: Breast cancer datasets ». Indian Journal of Computer Science and Engineering (IJCSE). 2. 756-763.

[59] Tadist, Khawla & Najah, Said & Nikolov, Nikola & Mrabti, Fatiha & Zahi, Azeddine. (2019). « Feature selection methods and genomic big data: a systematic review ». Journal of Big Data. 6.

[61] M. RAZA, N. D. Jayasinghe and M. M. A. Muslam, « A Comprehensive Review on Email Spam Classification using Machine Learning Algorithms », 2021 International Conference on Information Networking *(ICOIN)*, 2021, pp. 327-332.

2. Webographie

[2] Vajjala, S., Majumder, B., Gupta, A., & Surana, H. Text Classification. In Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems (Chapter 4).https://www.oreilly.com/library/view/practical-natural-language/9781492054047/ch04.html

[5] Kariuki, Charles. (2022, March 31). « Building a Multi-Class Text Classification Model using H2O and scikit-learn ».

https://www.section.io/engineering-education/building-a-multi-class-text-classification-model-using-h2o-and-sckit-learn/

[32] Nicholas, P., Tayaza, F., Kempa-Liehr, A. W., & O'Sullivan, J. M. (2022). « A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction », Front Bioinform,2. https://www.frontiersin.org/articles/10.3389/fbinf.2022.927312.

[40] https://www.obitko.com/tutorials/genetic-algorithms/search-space.php

[42] https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_fundamentals.htm

[43] Gad, Ahmed. « Introduction to Optimization with Genetic Algorithm »

.https://towardsdatascience.com/introduction-to-optimization-with-genetic-algorithm-2f5001d9964b

[54] https://www.geeksforgeeks.org/data-preprocessing-in-data-mining/

[55] Khanna, Chetna. « Text pre-processing: Stop words removal using different libraries ».

https://towardsdatascience.com/text-pre-processing-stop-words-removal-using-different-libraries-f20bac19929a

[62] https://www.python.org/doc/essays/blurb/

[63] Driscoll, M. Jupyter Notebook: An Introduction.
https://realpython.com/jupyter-notebook-introduction/

[64] https://www.tutorialspoint.com/scikit_learn/scikit_learn_introduction.htm

[65] https://www.tutorialspoint.com/python_data_science/python_matplotlib.htm

[66] https://www.tutorialspoint.com/python_data_science/python_pandas.htm

[67] Rouse, M. (2023, May 4). « Natural Language Toolkit ». https://www.techopedia.com/definition/30343/natural-language-toolkit-nltk

[68] NumPy: the absolute basics for beginners. NumPy. https://numpy.org/doc/stable/user/absolute_beginners.html

[69] Restaurant Reviews **:**Dataset for Natural language Processing. https://www.kaggle.com/datasets/d4rklucif3r/restaurant-reviews

[70] Fake and real news dataset **:**Classifying the news

https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset